AFWAL-TR-80-3125

AD A102145

LEVEL

BASIC RESEARCH IN DIGITAL STOCHASTIC MODEL ALGORITHMIC CONTROL

RAMAN K. MEHRA
JOHN S. ETERNO
RAMINE ROUHANI
ROBERT B. WASHBURN, JR.
DAVID B. STILLMAN
L. PRALY

SCIENTIFIC SYSTEMS, INC.
54 RINDGE AVENUE EXTENSION
CAMBRIDGE, MA 02140

NOVEMBER 1980

DTIC

S   D

JUL 29 1981

D

FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

DTIC FILE COPY

81 7 28 038

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.
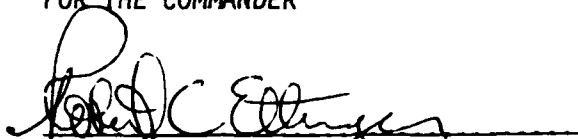
This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

ROBERT C. SCHWANZ, Project Engineer
Control Dynamics Branch
Flight Control Division

R. O. ANDERSON, Chief
Control Dynamics Branch
Flight Control Division

FOR THE COMMANDER

ROBERT C. ETTINGER, Colonel, USAF
Chief, Flight Control Division

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/FIGC, WPAFB, OH 45433 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
AFWAL-TR-80-3125	AD-A102145	

4. TITLE (and Subtitle)	5. TYPE OF REPORT & PERIOD COVERED
BASIC RESEARCH IN DIGITAL STOCHASTIC MODEL ALGORITHMIC CONTROL.	final report. May 1979 — Sept 1980
	6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)	8. CONTRACT OR GRANT NUMBER(s)
R. K. Mehra, J. S. Eterno, R. Rouhani, B. Washburn, Jr., D. B. Stillman, L. Praly	F33615-79-C-3613

9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
Scientific Systems, Inc. 54 Rindge Avenue Extension Cambridge, MA 02140	61102F 2304N312

11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE
AFWAL/FIGC Wright-Patterson AFB, OH 45433	November 1980
	13. NUMBER OF PAGES
	262

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report)
Same	Unclassified
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Digital control; impulse response; model following; adaptive control; self-tuning regulator; model reference adaptive; Model Algorithmic Control; stochastic control systems.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report presents the results of basic research in Model Algorithmic Control (MAC) with application to flight control systems. MAC is a new digital control design approach which relies on the following features: (i) an internal model of the system to be controlled; (ii) a reference trajectory description of desired closed loop behavior; and (iii) an on-line optimization of future control inputs to produce the desired performance.

MAC is well suited to the new generation of microprocessors, and is applicable to a wide variety of aerospace problems.

DD FORM 1 JAN 73 1473   EDITION OF 1 NOV 65 IS OBSOLETE

20. ABSTRACT (continued)

The purpose of this report is to explore the theoretical foundations and robustness of MAC and to examine its application to flight control problems. A missile attitude control simulation is chosen as a typical control design problem and used to demonstrate the application of MAC theory.

The major conclusion of this report is that MAC is a very flexible, powerful and promising technique for linear multivariable control design. Its theoretical properties of optimality, convergence, robustness and stability are verified by simulations of the missile attitude control example.

It is recommended that future studies of MAC concentrate on its extensions to the adaptive case and on flight test demonstration of MAC concepts.

FOREWORD

The research described in this report was performed by Scientific Systems, Inc., Cambridge, Mass. under Contract F33615-79-C-3613. The project was administered by Dr. Robert Schwanz in the Control Analysis Group of the Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio. Professor J. Gary Reid from AFIT participated in several stimulating discussions and provided the missile control example.

Dr. R. K. Mehra was the Principal Investigator. His co-workers were Dr. John S. Eterno, Dr. Ramine Rouhani, Dr. Robert B. Washburn, Jr., and Mr. David B. Stillman. Dr. Andre Rault and Mr. L. Praly of ADERSA/GERBIOS in France served as consultants on this project.

Accession For		
NTIS GRA&I	✗	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution/		
Availability Codes		
Dist	Avail and/or Special	
A		

TABLE OF CONTENTS

LIST OF FIGURES

NOMENCLATURE

The following nomenclature applies to all parts of the report except Section VIII and Appendix A. The notation for those is given within the section.

$a_{k,j}(i)$ impulse response from the $k^{\underline{th}}$ input u_k to $j^{\underline{th}}$ output y_j at lag i. For single-input single-output systems, the notation h_i is used to denote the impulse response at lag i.

$B_{ms}(z)$ stable factor of model transfer function $H_m(z)$

$B_{0s}(z)$ stable factor of system transfer function $H(z)$

$B_{0u}(z)$ unstable factor of $H(z)$

$c(t)$ desired set point value at time t

$C(z)$ z-transform of c(t)

$cov(t,\tau)$ covariance function between two variables at time t and τ

$\underline{d}$ impulse response error vector defined as $\underline{\tilde{h}}^T - \underline{h}^T$

e an error term

$\underline{\tilde{h}}$ model of identified impulse response for single-input single-output system, $(N+1) \times 1$ vector $[\tilde{h}_0, \ldots, h_N]$

$H(z), \tilde{H}(z)$ z-transforms of $\underline{h}$ and $\underline{\tilde{h}}$

$\inf\limits_{x}$ infimum over x (greatest lower bound)

J_T — weighted least squares performance index over time horizon T

m, M — lower and upper amplitude constraints on input $u(t)$

N — length of impulse response vector

p — prediction horizon

P — solution of Riccati equation for output matching (Section IV); or optimization weighting matrix (Section V)

q — scalar gain mismatch factor defined as $\underline{h} = q\hat{\underline{h}}$

$q(t)$ — linear weighting term (Section V)

R — mismatch matrix between impulse responses (Section III); or measurement noise matrix (Section VI)

$\mathbb{R}^N$ — N-dimensional Euclidean space

$\sup_x$ — supremum over x (least upper bound)

$u_f(s,t)$ — fictitious future input for time (t+s) based on computations at time t

$u_k(n)$ — k^{th} input at sample time n

$u^*(t)$ — optimal input at time t

$U(z)$ — z-transform of $u(t)$

v — coefficients of input function (Section V)

$\underline{v}$	measurement noise (Section VI)
$w(t)$	process noise at time t
w_k	non-negative weighting factor at lag k
$\underline{x}(t)$	state vector at time t
$y_j(t)$	actual value of $j\underline{th}$ output at sample time t
$y_M(t)$	model output value at time t
$\{y_p(\tau)\}_{t+1}^{t+T}$	predicted output trajectory from time (t+1) to time (t+T)
$y_p(s,t)$	predicted output at t+s based on observations up to time t
$y_r(t)$	reference output at time t
$\{y_r(\tau)\}_{t+1}^{t+T}$	reference output trajectory from time (t+1) to time (t+T)
$Y(z)$	z-transform of $y(t)$
y_{pzi}	zero-input response part of predicted output y_p
y_{pzs}	zero-state response part of predicted output y_p
$z(t)$	measured output at time t
z^{-1}	z-transform variable (or lag operator)

α	reference trajectory parameter, $0 \leq \alpha < 1$. Higher α denotes slower reference trajectory.
Ω	set of admissible input functions
σ^2	noise variance
σ_o^2	plant output variance for open loop prediction MAC
σ_c^2	plant output variance for closed loop prediction MAC
σ_w^2	variance of colored noise $w(t)$
ρ	one lag correlation coefficient of process noise (Section III); weighting factor for input in the quadratic performance index (Section VI)
ρ', ρ''	robustness indices
Φ	domain of robustness in Section III
Φ', ϕ'	control law functions (Section V)
ϵ_i	gain mismatch parameters
$\epsilon(t)$	tracking error $(y_r(t) - y(t))$
$\xi(t)$	a vector of inputs defined as $[u(t), u(t-1), \ldots, u(t-N-1)]^T$
Δ	sampling interval (seconds)

$E[y(t)|Y(t)]$ conditional expectation of $y(t)$ given $Y(t)$

$(\hat{})$ estimate of variable ( )

$(\underline{})$ vector of elements $(\ )_i$

$(\ )_{OL}$ open loop prediction of variable ( )

$(\ )_{CL}$ closed loop prediction of variable ( )

$(\ )_a$ partial derivative of variable ( ) with respect to variable a (Section VIII)

■   QED (end of proof)

SECTION I

INTRODUCTION AND SUMMARY

This report presents the results of basic research in Model Algorithmic Control (MAC) with application to flight control systems. MAC is a new digital control design approach which relies on the following features:

(i)   an internal model of the system to be controlled

(ii)   a reference trajectory description of desired closed loop behavior

and (iii)   an on-line optimization of future control inputs to produce the desired performance.

MAC is well suited to the new generation of microprocessors, and is applicable to a wide variety of aerospace problems.

The purpose of this report is to explore the theoretical foundations and robustness of MAC and to examine its application to flight control problems. A missile attitude control simulation was chosen as a typical control design problem and used to demonstrate the application of MAC theory. Section II presents a description of the MAC design philosophy for general systems and a discussion of its application to linear multi-variable problems in particular. Section III presents a mathematical formulation of an idealized discrete-time MAC and examines robustness, noise behavior, and the effects of input constraints. Section IV extends the theoretical analysis of MAC to non-minimum phase and time-delay systems. Section V examines a continuous-time MAC in order to provide a more general mathematical framework and application guidelines for the techniques. Section VI then discusses MAC design for continuous systems with discrete

1

noisy observations and unobserved outputs, and provides a formulation of the problem of sample rate selection. Section VII discusses improvements to the MAC predictor to handle process and measurement noise.

Next, the application of MAC and its software package IDCOM (for Identification and Command) to a missile control example is discussed in Section VIII. IDCOM is described functionally, and the missile simulation details are provided. The simulation results for several different test series are given in Section IX. Section X presents overall conclusions and recommendations for future work in this area.

The major conclusion of this report is that MAC is a very flexible, powerful and promising technique for linear multivariable control design. Its theoretical properties of optimality, convergence, robustness and stability are verified by simulations of the missile attitude control example. It is recommended that future studies of MAC concentrate on its extensions to the adaptive case and on flight test demonstration of MAC concepts.

SECTION II

MAC DESCRIPTION

Overview

The purpose of this section is to provide a general introduction to
MAC philosophy and to compare it qualitatively with other control design
techniques. The three main principles of MAC are discussed in Section 2.1;
the special features of MAC linear multivariable systems and a comparison
with other modern control techniques are given in Sections 2.2 and 2.3.
The remaining two sections (2.4 and 2.5) are devoted to a component
functional description of stochastic MAC and a comparison with Self-
Tuning Regulators and Model Reference Adaptive Control techniques.

2.1 MAC General Description

Some of the recurring problems in the application of modern control
theory to flight control systems are: (i) model selection, (ii) incor-
poration of state and control constraints, and (iii) robustness and sen-
sitivity to unknown parameters and disturbances.

Here we present a technique called MAC which treats the above prob-
lems in an effective manner. The technique was originally developed
for industrial applications in France. It is based on an identification-
optimization approach, which is very general in nature, and is a digi-
tal technique which makes full use of the capabilities and growth

potential of current microprocessors. The MAC strategy relies on the
following three features: 1) internal model of the system, 2) reference
trajectory and output constraints, and 3) control trajectory computa-
tion.

2.1.1   Internal Model of the System

The multivariable system to be controlled is represented by a mathe-
matical model in time-domain of the input-output type (see Figure 1).
For linear systems, the model is of the impulse response type, a repre-
sentation which has certain distinct advantages over the state space
representation or the transfer function representation for multivariable
control. For nonlinear systems, both the state space and the input-
output representations have certain advantages and disadvantages. In
applications, one may use either or both depending upon the nature of non-
linearities and the complexity of the resulting controller. The purpose
of the internal model is to have a flexible representation of the controlled
system stored in the computer memory, which can be updated as the system
changes and which can be used at any instant to predict the future be-
havior of the system under different control inputs. The internal model
of the system is used to compute optimal inputs, to detect process changes,
sensor malfunctions and severe faults. The inputs and current output
of the internal model are updated according to the actual observed values
of these variables, but any large difference between the computed and the
actual values gives important clues as to the malfunctioning of sensors
and actuators.
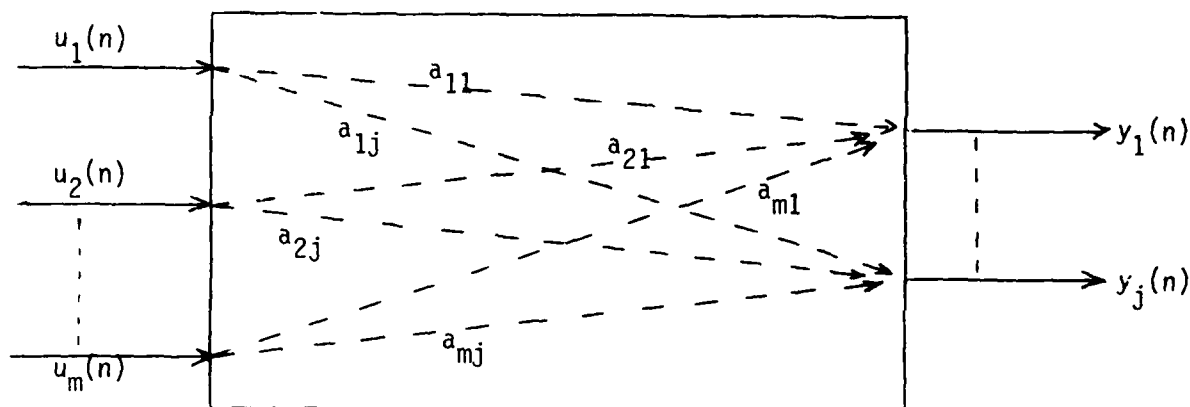
$\{a_{k,j}(i),\ i=1,\ldots,N\}$ represents the impulse response from $k\underline{\text{th}}$ input to $j\underline{\text{th}}$ output.

Figure 1. Impulse response representation of a linear system.

The internal model of the system is generally obtained via off-line identification, using either a physical model structure when one is easily available or an input-output representation model such as the impulse response model, which may change with the operating point. Some form of on-line parameter identification may also be done in those cases where large random variations of system parameters are expected. It has been found from experience that the robustness of MAC is sufficient to take care of small parameter changes.

2.1.2   Reference Trajectory and Output Constraints

The desired response of the closed-loop system is specified in the form of a reference trajectory and constraints are updated on-line using the actual output of the system. It is possible to handle output-dependent constraints[1] in this fashion and to eliminate the steady state measurement

---

[1]General state-dependent constraints can also be handled if an observer (or filter) is used to recreate the states from the measured outputs.

5

offset error. It should be noticed that the specification of reference

trajectories and constraints is much easier and natural than the specifi-

cation of a scalar performance index. Typically, the requirements on

controlled outputs are stated as "no overshoot," "fast response," or

"within maximum and minimum limits." These requirements are difficult to

express in a scalar quadratic performance index, but they are easily con-

verted into desired trajectories and constraints which can also be ex-

plained to the operators without any difficulty. In a hierarchical control

system, reference trajectories for lower levels may be specified by the

higher levels through an optimization process, since the controlled outputs

of lower levels are the control variables for the higher levels. The ref-

erence trajectories at the highest level would generally be obtained via

a combination of on-line and off-line optimization.

The concept of using reference trajectories is more general than

model-following. Firstly, it may not be possible to represent a reference

trajectory by a simple model, and secondly, under sensor or actuator

fault conditions, one may have to relax the system requirements to con-

trol within a band or tolerance limit. Certain control problems in-

volving more controlled outputs than control variables are formulated

more correctly as band-control problems rather than model-following or

scalar performance index problems.

2.1.3   Control Trajectory Computation

Controls are computed, in general, for a number of future time points

using an iterative optimization technique which minimizes the distance

6

between the desired reference trajectory and output trajectory predicted by the internal model, while keeping all the output, state and control constraints satisfied. The complexity of the control algorithm is directly dependent on the structure of the internal model, the number of inputs and outputs and the constraints. For linear systems, the impulse response representation results in a simple, fast, projection-type quadratic programming solution which can be implemented in micro/mini-computers of the present generation. The actual dimensionality of the state does not increase the complexity of the algorithm as it would in a state vector representation.

2.2 Special Features of MAC and IDCOM (IDentification and COMmand) for Linear Multivariable Control

The MAC design approach has been found to be flexible and robust. It is well suited for the evolving microprocessor technology providing high speed memory and fast computation times for basic calculations such as convolutions. The universality of the impulse response representation leads to a unified design approach for systems of all orders. Furthermore, the parameter-linearity of this representation leads to a duality between identification and control.

MAC is implemented by a computer program called IDCOM. The special features of IDCOM are: (i) no model order reduction is required since an impulse response representation is used. (ii) Input magnitude and rate constraints are handled directly and exactly. (iii) The control law is time-varying and the closed loop response is robust to parameter changes. (iv) Gain scheduling is replaced by on-line updating of the internal model

using operating data and parameter estimation techniques, thereby reducing reliance on theoretical models of the system. (v) The same algorithm is used for impulse response identification and for control law computation, thereby simplifying the hardware requirements. (vi) The control laws can be modified on-line in case of sensor failures or degraded system performance.

2.3 Comparison of IDCOM with Other Modern Control Techniques

IDCOM, like most other modern control techniques, is a multivariable time-domain technique. It is different, however, in several important ways. These differences are important since they account for the success of IDCOM in control applications where the lack of good models has prevented the use of other modern control techniques (Richalet et al., 1978). The important differences are:

(i) Robustness. Modern control techniques generally use full state feedback based on a parametric state vector model. IDCOM, on the other hand, uses output feedback based on an impulse response model. It can be argued that the effect of modeling errors on IDCOM will be less compared to that on state vector techniques.

(ii) Implementation. Modern control techniques based on state vector models generally require the solution of matrix Riccati equations. Since these equations are computationally time-consuming to solve for systems of high order, the practical implementation of these techniques requires model reduction and off-line solution of the Riccati equation. In practice, only the steady state gains from the Riccati equation are implemented and these gains are scheduled as a function of the operating point

8

(e.g., flight condition). This implementation does not allow for on-line changes in system model and performance criteria. The IDCOM approach, on the other hand, is much more flexible and adaptive, since the model, criteria, and sampling rates can be adjusted on-line. This flexibility comes from the use of the impulse response representation, which has the additional advantages of conceptual simplicity, ease of identification and elimination of the model order reduction problem.

(iii) <u>Criterion Function and Constraints</u>. Most of the modern control techniques require specification of a scalar criterion function. In practice, this is very difficult or impossible to do. The specification of the information used in IDCOM is easier and more natural. Basically, the specification of desired responses and constraints is simpler than the specification of a scalar performance index. It is also possible to set up a priority list on the control of outputs in IDCOM and even to make them conditional on the occurrence of some future events. For example, the failure of a sensor can be detected and the control strategy can be shifted accordingly. The problems of sensor blackout or computer overload can also be handled since controls are computed and stored for several future time points and these can be used till new information or computational capability becomes available. The exact satisfaction of control and output constraints is absolutely essential in many applications. Such constraints are handled much more easily in IDCOM, thanks to the impulse response representation of the system.

(iv) <u>Duality of Identification and Control</u>. From a practical standpoint, the duality of identification and control for an impulse response model is much more useful than the duality between estimation and control

9

for state vector models. The former can be used directly to increase the robustness of the controller by on-line identification whenever the residuals between the model output and the system output are consistently outside some prespecified confidence bands.

Some of the modern control approaches which are closer to MAC are the Model Reference Adaptive (MRA) (Landau, 1974) and the Self-Tuning Regulator (STR) (Astrom, 1980) approaches. They are, however, not exactly similar since the specification of reference models and the computations of control are done differently, as discussed in Section 2.5 below.

2.4 Component Functional Description

In order to explain the operation of MAC and compare it to other approaches, it is useful to describe the functions of each explicit MAC component. Figure 2 shows different elements of a fairly general MAC scheme consisting of seven different functional blocks. These are:

1. Internal model of the plant which is meant to be a close representation of the system (this model may be in state vector form, linear or nonlinear, impulse response form, transfer function form or even table look-up, depending on the application).

2. Reference trajectory generator, which uses the latest plant outputs and desired set point conditions to generate desired reference trajectories for the controlled closed loop system to follow.

3. Control signal generator performs most of the main computations within MAC to produce control signals for the actuators. The inputs to this block consist of the predicted error between the output of the
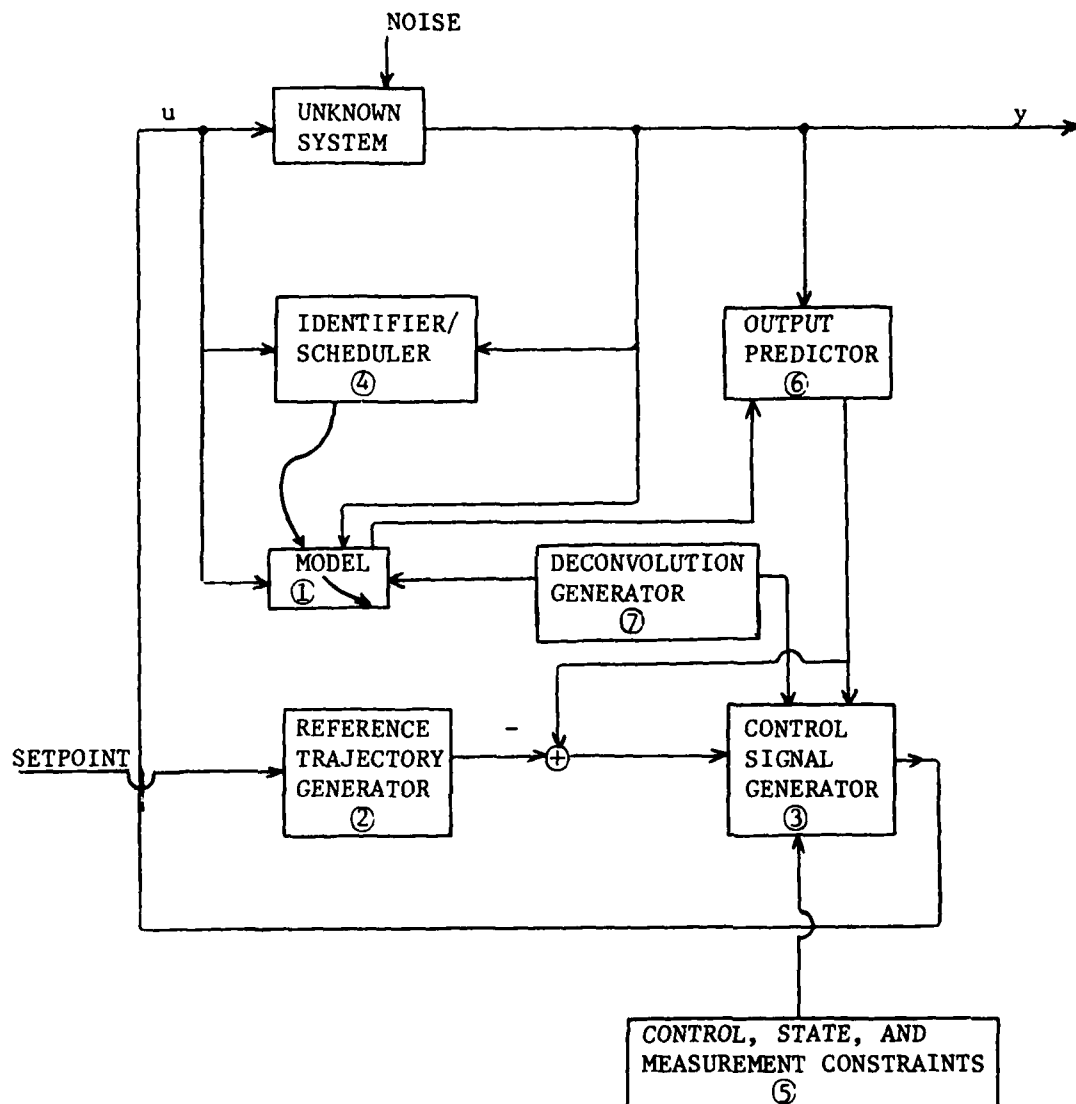
10

SELF TUNING REGULATOR (STR)		MODEL REFERENCE ADAPTIVE (MRA)		MAC
IMPLICIT	EXPLICIT	IMPLICIT	EXPLICIT	EXPLICIT
① + ③ + ④	①	①	①	①
	③	② + ③	②	②
	④		③	③
				④
				⑤
				⑥
				⑦

Figure 2. Relation of adaptive MAC to other adaptive control schemes.

11

reference trajectory generator, ②, and the output predictor, ⑥, along with the control and state constraints, ⑤.[2]

The control algorithm is iterative involving several trials of different control inputs to calculate the best input for minimizing the tracking error without overstressing the actuators and the computational facilities of the process control computer. The model used for control computations may not be the same as the internal model, ①, of the plant.

4. <u>Identifier/scheduler</u> performs the function of updating the internal model ① of the plant under changing operating and plant conditions. If the changes in the plant can be expressed as a deterministic function of the operating state, off-line or periodic identification followed by scheduling will be adequate.

However, if the changes in the plant are rapid and random, on-line identification is required, which may be passive or active. In passive on-line identification, no extra signals are injected into the system to enhance identifiability or to improve the speed of identification. In active on-line identification, extra input signals are used to enhanc identifiability in order to match the frequency of adaptation with the frequency of plant variations (see Mehra <u>et al</u>., 1978).

Input signal design techniques (Mehra, 1980) play an important role in active adaptation. The identifier can also be augmented to perform Fault Detection, Diagnosis and Prognosis (DDP) functions.

5. <u>Control, state, and measurement constraints</u>, which are generally of the amplitude and rate type, may in some cases (e.g., mixed state and control constrain s) be much more complicated.

---

[2] Circled numbers refer to blocks in Figure 2.

6. <u>Output predictor</u> is an important element of MAC since the accuracy of plant predictions contributes directly to the performance of any control system. The predictor may be of the state type (Kalman, extended Kalman, etc.), Luenberger observer type, impulse response, or ARMA (Box and Jenkins, 1976 or Astrom, 1970). The latter type is common in Self-Tuning Regulator (STR) and Model Reference Adaptive (MRA) control applications. They are also discussed in Section VII.

7. <u>Deconvolution generator</u> is used in those cases where a direct use of the internal model will lead to undesirable control behavior, e.g., nonminimum phase or time delay systems. Several techniques for generating deconvolution models are discussed in Section IV and Appendix A.

2.5 <u>Comparison with STR and MRA</u>

In Figure 2, a comparison of MAC, STR, and MRA is given in terms of function blocks used in each scheme. For example, in STR with implicit plant model, functions ①, ③, and ④ are combined into one block, i.e., the controller directly identifies the gains of the control law without explicit identification of the plant model.

In this scheme, blocks ②, ⑤, ⑥ and ⑦ are either absent or they are combined with ①, ③ and ④. Such a combination results in a simple design at the expense of flexibility. For example, one cannot change on-line reference trajectories or account for failed sensors. In explicit STR, an ARMA model is identified on-line and then used to develop the control law. This scheme is more suitable for nonminimum phase systems, but still cannot handle constraints exactly.

MRA may be implicit or explicit with respect to the reference model. The difference is indicated in Figure 2, where ② + ③ implies that functions ② and ③ are combined in the implicit MRA technique.

The obvious difference between MAC and the other techniques is separation of different functions. In some practical applications, all the functions may not be required, but the great generality, robustness and flexibility of MAC come from having all these functions in the software. With modern advances in digital hardware, one incurs very little cost for having separate functions, whereas the payoff in terms of performance and flexibility of the control scheme can be quite high.

2.6 Summary

In this section, the philosophy and basic concepts of MAC were discussed. The similarities and differences between MAC and other modern control techniques were outlined, and the generality and flexibility of MAC for control design purposes were emphasized.

SECTION III

MATHEMATICAL FORMULATION AND ROBUSTNESS OF IDEALIZED DISCRETE-TIME MAC

Overview

The purpose of this and some of the subsequent sections is to develop a mathematical framework for an analytical study of MAC. The description of MAC in Section II reveals that a great deal of intuition is involved in MAC design.[3] Therefore, a complete mathematical analysis of MAC properties is extremely difficult. Several simplifying assumptions are required to make the analysis tractable. The purpose of mathematical analysis is to serve as a guide for MAC design rather than a replacement for it. In this section, we analyze unconstrained linear single-input single-output discrete-time, minimum phase systems. Both deterministic and stochastic (colored output additive noise) systems are considered. The robustness problem is discussed and performance "measures" of robustness are proposed. Some of results presented here are more generally applicable to optimizing type of control laws of which MAC is one example. In particular, a close link is established between the stability and closed loop properties of optimizing type and feedback type of control laws.

The organization of this section is as follows. Section 3.1 presents an overview of the major functions of MAC (prediction-optimization). Section 3.2 treats the case of a single input-single output system where the inputs are free of constraints, and introduces linear closed loop and open loop predictors. Section 3.3 is concerned with the stability analysis,

---

[3]The reader may be interested in looking at Sections 8.1, 8.2 and 8.3 for further details of the MAC algorithm.

15

both from a time domain and frequency domain viewpoint. Section 3.4 investigates the effect of an output additive noise and compares the performances of the control with respect to reference trajectories under open and closed loop prediction schemes. Section 3.5 poses the robustness problem and its relation to stability. Section 3.6 defines two performance indices for robustness analysis. Finally, in Section 3.7 constraints on input are introduced and the analysis of the constrained system is related to the analysis of the unconstrained one.

3.1 Mathematical Formulation

As discussed in Section 2.5, MAC is conceptually similar to a model reference adaptive type of control with some important differences in practical implementation. It involves (i) dynamic models for system representation and prediction, (ii) a reference trajectory and (iii) an optimality criterion leading to the optimal control. A schematic representation of MAC (Figure 3) displays the main operations performed: (i) prediction of the future output for T steps ahead based on the output $(y(t))$ of the actual plant and on the computed output $(y_M(t))$ of the plant's model, (ii) calculation of the future reference trajectory based on the actual output $(y(t))$ of the plant and a desired set point $c(t)$, and (iii) computation of the input $u(t)$ based on the above predictions and using a certain optimization criterion.

3.1.1 Representation and Prediction

The system is represented by its impulse responses, the identification of which can be done both on-line and off-line. However, in most cases the off-line identification is accurate enough for the purpose of control

16
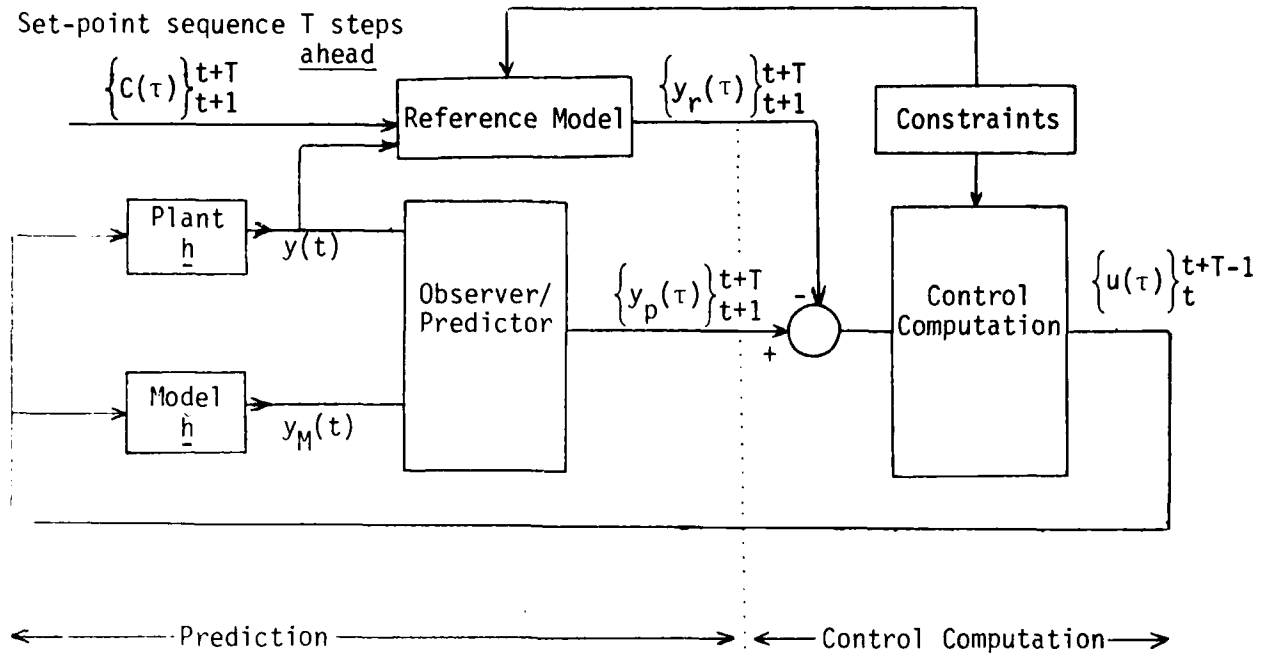
Figure 3. Model Algorithmic Control.[4]

and one can avoid the cost and complexity of an on-line identification procedure. This is due to the particular redundancy of the impulse response representation which allows a considerable enhancement of the robustness of the control scheme against identification errors and parameter perturbations (Mehra et al., 1979; Mereau et al., 1978).

The formal representation of the system is as follows:

$$y(t+1) = \underline{h}^T \underline{u}(t) = h_0 u(t) + h_1 u(t-1) + \ldots + h_N u(t-N) \qquad (3.1)$$

where $y(t+1)$ is the plant output at time $t+1$; $\underline{h}^T \in \mathbb{R}^{N+1}$ denotes the plant impulse response; $u(t-j) \in \Omega \subset \mathbb{R}$ for $j = 0,\ldots,N$. $u(t-j)$ is the input at time $(t-j)$ to the plant. Ω is the constraint set of the input.

The model (3.1) is also called the Actual Model to emphasize that $\underline{h}^T$ represents the actual impulse response. But since such perfect knowledge

---

[4] $\left\{c(\tau)\right\}_{t+1}^{t+T}$ denotes the sequence $\{c(t+1),\ldots,c(t+T)\}$.

17

of the plant impulse response is usually not possible, one has to use an approximation $\tilde{\underline{h}}^T$ to $\underline{h}^T$. The model corresponding to this latter impulse response is then:

$$y_M(t+1) = \tilde{\underline{h}}^T\underline{u}(t) \tag{3.2}$$

The above model, together with the past history of the plant output denoted by $Y(t) = \{y(\tau), \ \tau \leq t\}$ is used to predict the future value of the output. Various prediction schemes are conceivable. In this section we will limit ourselves to simple open loop and closed loop prediction. Figure 4, where $y_p(t)$ refers to the predicted value of the output, summarizes the representation-prediction part of MAC.
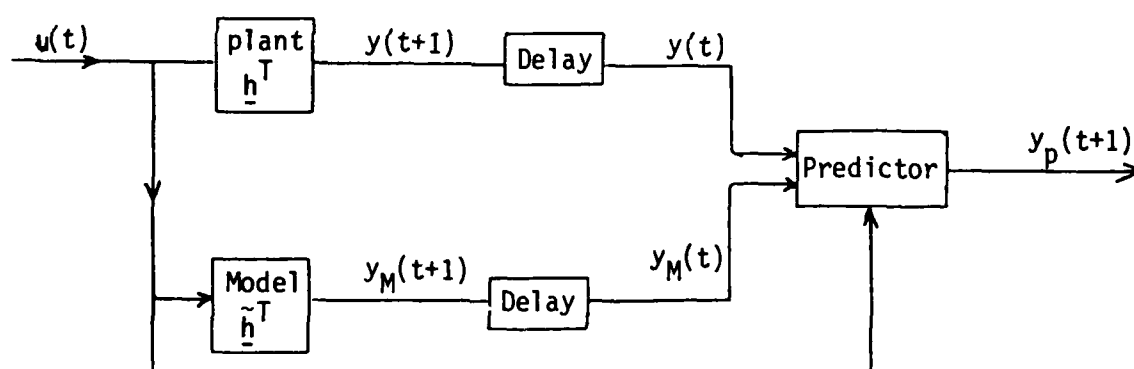
Figure 4. MAC prediction.

3.1.2 The Reference Trajectory

The purpose of the control is to lead the output $y(t)$ along a desired, and generally smooth, path to an ultimate set point C. Such a path is called a reference trajectory. In the present section the reference trajectory is of first order and is initiated on the output of the plant at

18

time t_0:

$$y_r(t_0+k) = \alpha^k y_r(t_0) + (1-\alpha^k)c \quad k = 1,\ldots,T$$

$$y_r(t_0) = y(t_0)$$

(3.3)

The reference trajectory can be chosen to be of higher order and the set point c can be made time varying. These latter extensions would not bring any conceptual difficulty in the solution of the control problem, although the actual computation of the inputs becomes more complex, but not intractable, as the order of the reference trajectory increases.

3.1.3 The Optimality Criterion and the Optimum Control Strategy

The optimality criterion should reflect the previously mentioned purpose of following the reference path to the desired set point c. That can be done by defining the optimum control strategy as the one which minimizes over a certain horizon in the future, the deviation of the predicted outputs from the reference path. Formally, at each instant t_0, the optimum set of T future inputs $\{u*(t_0), u*(t_0+1),\ldots, u*(t_0+T-1)\}$ are such that the predicted T outputs $\{y_p(t_0+1),\ldots,y_p(t_0+T)\}$ are as close as possible, in the sense of a weighted Euclidean norm, to the reference trajectory y_r. Therefore the function to minimize is:

$$J_T = \sum_{k=1}^{T} (y_p(t+k) - y_r(t+k))^2 w_k$$

(3.4)

w_k : nonnegative weighting factor

Note that at time t_0, the determination of T optimal inputs $u*(t_0+k)$

19

$(k = 0, \ldots, T-1)$ is done by solving a static optimization problem:

$$\text{Minimize } J_T(u(t_0), \ldots, u(t_0+T-1))$$

$$\text{s.t.} \quad u(t_0+k) \in \Omega \text{ for } k = 0, \ldots, T-1 \tag{3.5}$$

Various types of algorithms can be used to solve (3.5) and to determine the set $U^*(t_0) = \{u^*(t_0), \ldots, u^*(t_0+T-1)\}$ (Luenberger, 1973). The interval $\{t_0, t_0+T\}$ is called the horizon of control evaluation, and sometimes the horizon of prediction since at time t_0 one has to predict a set of T outputs $y_p(t+k)$, $(k = 1, \ldots, T)$. Once the set $U^*(t_0)$ of the T optimum inputs is determined, it is possible to wait up to T periods before observing the actual output $y(t)$, reinitializing $y_r(t)$, predicting y_p's and computing the next set $U^*(t_0+T)$. This means that all the elements of the optimum control set $U^*(t_0)$ have been actually applied to the plant.

A more appealing strategy consists of applying the first few optimal inputs $\underline{u}^*(t+k)$, $(k = 0, 1, \ldots, p$ with $p \ll T)$ before reinitializing y_r and computing the next T optimum inputs. In the limit, if the computing facilities allow, one would apply only the first optimum input $\underline{u}^*(t)$ of the set $U^*(t)$, observe $y(t+1)$, initialize the reference trajectory y_r on the observed value $y(t+1)$ and solve the optimization problem (3.5) at time $t+1$. In this latter case the unused optimum inputs $u^*(t+1), \ldots,$ $u^*(t+T-1)$ of $U^*(t)$ (computed at time t) will serve as starting values for the numerical algorithm determining $U^*(t+1)$, that is, solving (3.5) at time $(t+1)$. Only this latter case will be considered in this section. Figure 5 displays the optimum inputs and the corresponding predicted output over the horizon T. To visualize the overall control procedure

corresponding to the last mentioned case, one must repeat the same picture for t+1, t+2,.... Figure 6 is a block representation of the whole control scheme.
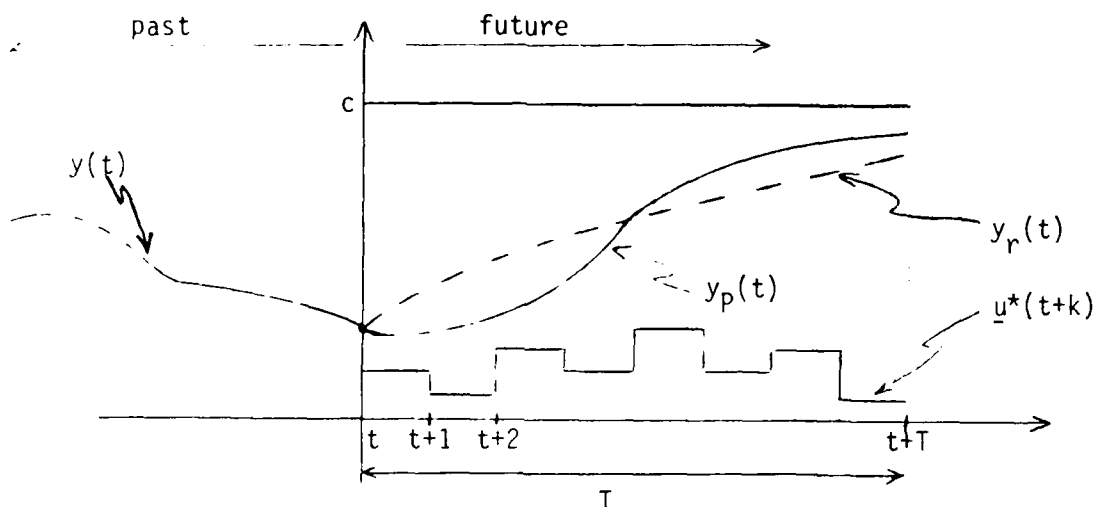
Figure 5.   MAC inputs and outputs.
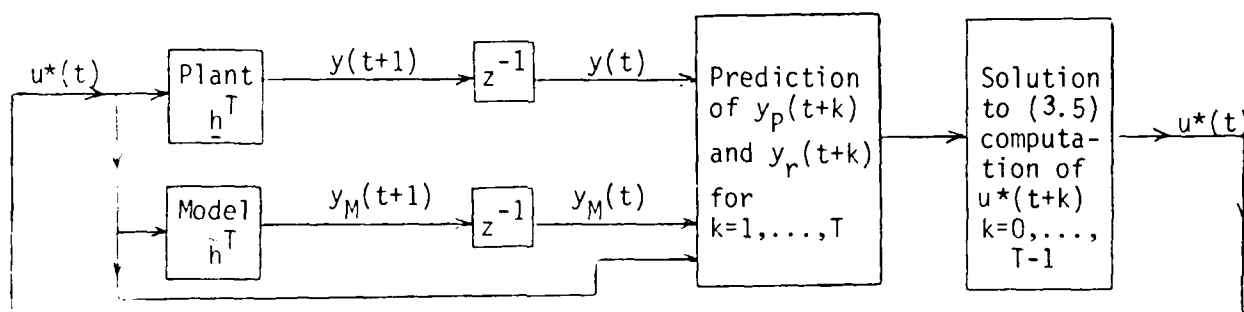
Figure 6.   Control block representation.

21

3.2 Unconstrained Input

3.2.1 Linear Prediction

So far the characterization of the prediction scheme has been kept general. Let us constrain the class of predictors to be linear. Notwithstanding the linearity of the plant, the above class of predictors is a practically convenient choice. Moreover, it simplifies substantially the determination of the optimum control sequence. Indeed, the linear character of $y_p(t)$ causes the function J_T to be quadratic in the inputs $u(t)$, and hence to become a candidate for fast quadratic optimization algorithms.

3.2.2 One Step Prediction (T = 1)

The assumption that the input vector $\underline{u}(t)$ is free of constraints, i.e., $\Omega \equiv R^N$, results in a significant simplification in the optimum control determination. That is, the length T of the horizon of prediction does not affect the optimum value of the first input $u^*(t)$ to be applied. In other terms, the minimization of J_T and $J_{T'}$, with $T' \neq T$, will result in the same first input $u^*(t)$ of the optimum sequences of length T' and T. In particular, the first element of the sequence $\{u^*(t),\ldots,u^*(t+T-1)\}$ minimizing J_T is identical to the input minimizing J_1. This is an important simplification, since it reduces a T-variable minimization at each step to a one-variable minimization. This property, which is based essentially on the principle of superposition of linear systems, is easily established, as the proof of the following proposition demonstrates.

Proposition 3.1. If the input to the plant defined by (3.1) is free of constraints, the minimization of J_T (3.4) and the minimization of J_1 at time

t_0, result in the same first input $u^*(t_0)$. That is, the first element of the set $U^*(t_0)$ corresponding to the minimization of J_T is identical to the input minimizing J_1.

Proof

$$J_T = \sum_{j=1}^{T} \|y_p(t_0+j) - y_r(t_0+j)\|^2 w_j, \quad w_j > 0$$

The value zero for J_T is achieved if and only if:

$$y_p(t_0+j) = y_r(t_0+j) \qquad j = 1,\ldots,T \tag{3.6}$$

The right-hand side of equation (3.6) depends on the observed value $y(t_0)$, α and c through (3.3), and hence is known independently of the input $u(t_0), u(t_0+1),\ldots$. The left hand side $y_p(t_0+j)$ depends linearly on $y_M(t_0+j)$ and on the observed value $\{y(\tau), \tau \le t_0+j-1\}$. Therefore, the inputs $u(t_0), u(t_0+1),\ldots,u(t_0+j-1)$ enter linearly in the expression of $y_p(t_0+j)$. The determination of the optimum input sequence is done as follows. For $j = 1$ the equation

$$y_p(t_0+1) = y_r(t_0+1) \tag{3.7}$$

determines uniquely $u^*(t_0)$. The next equation, corresponding to $j = 2$:

$$y_p(t_0+2) = y_r(t_0+2)$$

in which the value of $u(t_0)$ is set equal to the optimum $u^*(t_0)$, determines uniquely $u^*(t_0+1)$. Hence the optimal sequence $\{u^*(t_0),\ldots,u^*(t_0+T-1)\}$

23

is determined sequentially. The existence and uniqueness of a solution

for each equation of (3.6) is assured because of the linearity of $y_p(t+j)$.

Now let us consider the cost J_1:

$$J_1 = \| y_p(t_0+1) - y_r(t_0+1) \|^2$$

Clearly, the minimum of J_1 is achieved for:

$$y_p(t_0+1) = y_r(t_0+1) \tag{3.8}$$

which is identical to (3.6), resulting in the same solution $u^*(t_0)$. ∎

With respect to the above proof we should mention that:

(1) The linear character of $y_p(t+j)$ is a sufficient condition for

the proof to hold. If the predictor $y_p(t+j)$ is nonlinear and its depen-

dence on the past inputs and the observed data is denoted via a nonlinear

operator F:

$$y_p(t+j) = F(u(t+j-1), u(t+j-2),\ldots; y(\tau), \tau \leq t+j-1)$$

then the necessary general condition for the proof of proposition 3.1 to

hold is that

$$F(\cdot,u(t+j-2),\ldots,u(t+j-N-1); y(\tau), \tau \leq t+j-1)$$

must be a mapping of R onto R. In the ase of a nonlinear predictor, the

optimal input sequence is generally not unique.

(2) The T optimal inputs $u^*(t_0),\ldots,u^*(t_0+T-1)$ obtained by minimizing

J_T are not identical to the input sequence obtained by T consecutive mini-

mizations of J_1. This is partly because of the right-hand side of

equation (3.6). Indeed, while minimizing the distance over a horizon of length T, $y_r(t_0+j)$ is initialized on $y(t_0)$ for the whole length of interval T; but in the case of T consecutive minimizations of J_1 at time t_0, $t_0+1,\ldots,t_0+T-1$, the reference trajectory y_r is continuously initialized on the observed $y(t_0),y(t_0+1),\ldots,y(t_0+T-1)$; that is:

$$y_r(t_0+j) = \alpha y(t_0+j-1) + (1-\alpha)c \quad \text{for all } j = 1,\ldots,T$$

3.2.3   Open Loop and Closed Loop Prediction

In this section two simple one step prediction (T = 1) schemes, one open loop and the other closed loop, are investigated. The input is assumed to be free of constraints.

The simplest one step open loop predictor that one can imagine is:

$$y_p(t+1) = y_M(t+1) = \tilde{\underline{h}}^T\underline{u}(t) \tag{3.9}$$

that is, the model of the plant is used to predict the output one step ahead. The main inconvenience of this open loop predictor is, as one might expect, that the output $y(t)$ of the controlled plant would not converge to its desired final value c. This is easily established as follows. The optimum input sequence $\underline{u}(t)$, in the sense of minimizing J_1, satisfies (3.8), which together with (3.3) and (3.9) implies:

$$y_M(t+1) = \underline{h}^T\underline{u}(t) = y_r(t+1) = \alpha y(t) + (1-\alpha)c \tag{3.10}$$

from which we can deduce the equation defining the optimum input sequence as:

$$\tilde{h}^T \underline{u}(t) - \alpha \underline{h}^T \underline{u}(t-1) = (1-\alpha)c \qquad (3.11)$$

Now assuming that the above recursive equation is convergent (i.e., the system is stable), the equilibrium input u^* and plant output y^* are given as:

$$u^* = \frac{(1-\alpha)c}{\tilde{h}^T \underline{1} - \alpha \underline{h}^T \underline{1}}, \qquad \text{a constant scalar}$$

$$y^* = \underline{h}^T \underline{1} \left(\frac{(1-\alpha)c}{\tilde{h}^T \underline{1} - \alpha \underline{h}^T \underline{1}} \right), \qquad \text{a constant scalar} \qquad (3.12)$$

$$\underline{1}^T = (1,1,\ldots,1) \in \mathbb{R}^{N+1}$$

Since in general the model impulse response $\tilde{h}^T$ is different from the plant one $\underline{h}^T$, the ultimate value y^* differs from the desired value c. It is seen that the necessary and sufficient condition resulting in a nonbiased open loop prediction, i.e., $y^* = c$, is $\tilde{h}^T \underline{1} = \underline{h}^T \underline{1}$. But such a condition is more of academic interest and has little chance of holding in practical situations. However, it is important to note that the bias

$$\left(\frac{\underline{h}^T \underline{1} - \tilde{h}^T \underline{1}}{\tilde{h}^T \underline{1} - \alpha \underline{h}^T \underline{1}} \, c \right)$$

can be estimated since $\tilde{h}^T$ is known and $\underline{h}^T \underline{1}$, i.e., the gain of the process, can be estimated from the step response of the plant.

To overcome the above bias problem, one uses a closed loop prediction, a simple case of which is:

$$y_p(t+1) = y_M(t+1) + (y(t) - y_M(t))$$

$$= y(t) + (y_M(t+1) - y_M(t)) \qquad (3.13)$$

26

where $(y(t) - y_M(t))$ represents a correction term which is similar to the innovation term of a Kalman filter. Then the optimum control, by (3.7), is such that:

$$\alpha y(t) + (1-\alpha)c = y(t) + (y_M(t+1) - y_M(t)) \tag{3.14}$$

Now letting $t \to \infty$, and assuming that the system is stable (i.e., $\lim_{t \to \infty} y(t) < \infty$), one deduces:

$$\lim_{t \to \infty} y(t) = c$$

that is, the desired set point c is reached asymptotically. The previous bias has vanished.

3.3 Stability

In the previous section, it has been assumed that the optimally controlled system (both with open loop and closed loop prediction) is stable. This section addresses the stability condition both in time domain and in z-transform domain.

In both cases, the optimum sequence of inputs $u^*(t)$,is generated by an autoregressive equation, which results from (3.8) where $y_p(t+1)$ and $y_r(t+1)$ are expressed in terms of inputs:

open loop: (from equations (3.9) and (3.10))

$$\tilde{h}_0 u^*(t) = \alpha \sum_{j=0}^{N} h_j u^*(t-j-1) - \sum_{j=1}^{N-1} \tilde{h}_j u^*(t-j) + (1-\alpha)c$$

$$y(t) = \underline{h}^T \underline{u}^*(t-1) \tag{3.15}$$

<u>closed loop</u>: (from equations (3.9) and (3.14))

$$\tilde{h}_0 u^*(t) = \sum_{j=0}^{N-1} \tilde{h}_j u^*(t-1-j) - \sum_{j=1}^{N-1} \tilde{h}_j u^*(t-j) + (1-\alpha)[c - \sum_{j=0}^{N-1} h_j u^*(t-j)]$$

(3.16)

$$y(t) = \underline{h}^T u^*(t-1)$$

Obviously, if the sequence $u^*(t)$ tends to an equilibrium value u^*, that is, if the corresponding autoregressive model is stable, then the output $y(t)$ tends to an equilibrium value $y(\infty)$ which equals c in the case of closed loop prediction but differs from c for open loop prediction. However, the converse is not true; that is, theoretically one may have a converging output $y(t)$, while the input $u^*(t)$ diverges. Intuitively it is clear that even though $|u^*(t)|$ might increase indefinitely, the linear function $\underline{h}^T u^*(t)$ may remain finite (at least in theory). This is basically what happens in nonminimum phase systems. Such behavior is not acceptable in applications because of the nonrealizability of infinite inputs (Astrom, 1970; Astrom and Wittenmark, 1974).

The boundedness condition for the input sequence $u^*(t)$ is identica' to the stability of the autoregressive models, that is, the polynomials:

$$z[\tilde{h}_0 + \tilde{h}_1 z^{-1} + \ldots + \tilde{h}_{N-1} z^{-N-1}] - \alpha[h_0 + h_1 z^{-1} + \ldots + h_{N-1} z^{-N-1}] = 0$$

(open loop)   (3.17)

and

$$(z-1)[\tilde{h}_0 + \tilde{h}_1 z^{-1} + \ldots + \tilde{h}_{N-1} z^{-N-1}] + (1-\alpha)[h_0 + h_1 z^{-1} + \ldots + h_{N-1} z^{-N-1}] = 0$$

(closed loop)   (3.18)

must have all their roots within the unit circle.

28

So far we have mainly focused on the relationship between the output $y(t)$ and the optimum input sequence $u^*(t)$. Now let us turn our attention to the response of an optimally controlled system to the set point c, that is, the relationship between $y(t)$ and $c(t)$. In terms of the z-transforms, one deduces from (3.11) and (3.14) with some manipulations:

open loop:

$$\frac{U(z)}{C(z)} = \frac{(1-\alpha)}{H(z) - \alpha z^{-1}H(z)} \tag{3.19}$$

$$\frac{Y(z)}{C(z)} = \frac{z^{-1}H(z)(1-\alpha)}{\hat{H}(z) - \alpha z^{-1}H(z)} \tag{3.20}$$

closed loop:

$$\frac{U(z)}{C(z)} = \frac{(1-\alpha)}{z^{-1}(1-\alpha)H(z) + (1-z^{-1})\tilde{H}(z)} \tag{3.21}$$

$$\frac{Y(z)}{C(z)} = \frac{z^{-1}H(z)(1-\alpha)}{z^{-1}(1-\alpha)H(z) + (1-z^{-1})\tilde{H}(z)} \tag{3.22}$$

with $U(\cdot)$, $Y(\cdot)$, $H(\cdot)$, $\tilde{H}(\cdot)$ and $C(\cdot)$ denoting respectively the z-transforms of $u(t)$, $y(t)$, $h(t)$, $\tilde{h}(t)$ and $c(t)$. Figures 7 and 8 display the open-loop and the closed-loop cases.
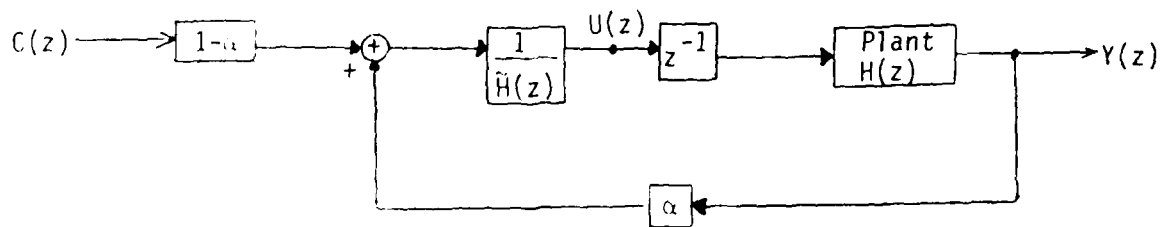
Figure 7. Open loop prediction.

29
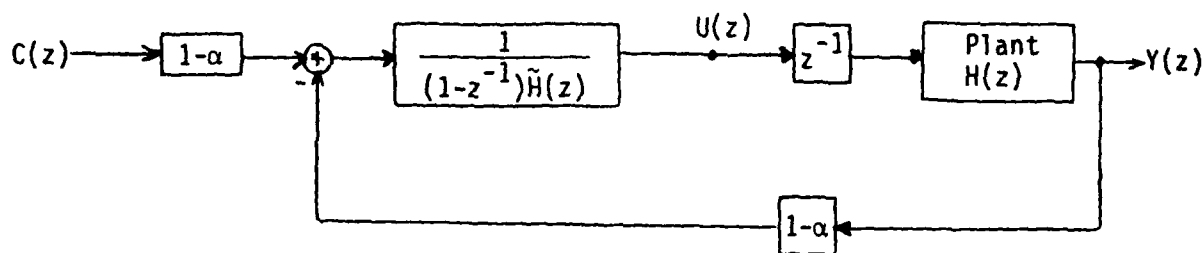
Figure 8. Closed loop prediction.

Let us first note that for a perfect identification of $\underline{h}$, that is, for $\tilde{H}(z) = H(z)$, the open loop and closed loop transfer functions become identical and given by

$$\frac{U(z)}{C(z)} = \frac{(1-\alpha)}{(1-\alpha z^{-1})H(z)} \tag{3.23}$$

$$\frac{Y(z)}{C(z)} = \frac{1-\alpha}{z-\alpha} \tag{3.24}$$

Note that under such perfect identification the transfer function of $Y(z)$ with respect to $C(z)$ is of first order and identical to the reference trajectory. Now if the plant itself is not stable, i.e., if the poly-nomial $H(z)$ has some of its roots outside the unit circle, then $U(z)$ corresponds to an increasing sequence $u(t)$. This latter case characterizes nonminimum phase systems where the cancellation of $z^{-1}H(z)$ in the original expression of $\frac{Y(z)}{C(z)}$, leading to (3.10), is not valid because $H(z)$ contains unstable zeros.

Let us come back to the case of imperfect identification, $\tilde{H}(z) \neq H(z)$. From the transfer functions (3.19) - (3.22) it becomes clear that for the system to be stable, i.e., the output $y(t)$ to be convergent and the opti-mum input sequence to be bounded, it is necessary and sufficient that:

open loop: $\tilde{H}(z) - \alpha z^{-1}H(z)$ has all its roots within the unit circle

closed loop: $z^{-1}(1-\alpha)H(z) + (1-z^{-1})\tilde{H}(z)$ has all its roots within the unit circle

It is easy to verify that the above polynomials are characteristic polynomials of the autoregressive models (3.17) and (3.18).

Note that in general the exact plant transfer function $H(z)$ is not known and therefore one cannot evaluate the exact expression of the above polynomial. However, if the identification of $H(z)$ is fairly good, then it is expected, by the continuity theorem (which states that the roots of a polynomial are continuous functions of its coefficients), that the roots of:

$$\tilde{H}(z) - \alpha z^{-1}H(z) \qquad \text{(open loop)}$$

and $\qquad z^{-1}(1-\alpha)H(z) + (1-z^{-1})\tilde{H}(z) \quad \text{(closed loop)}$

are close to the roots of:

$$(1-\alpha z^{-1})\tilde{H}(z)$$

Therefore, the stability of the identified model $\tilde{H}(z)$ implies the stability of the system. But when the identification error becomes large so that the discrepancy between $\tilde{H}(z)$ and $H(z)$ becomes significant, then to determine the stability of the system one should have recourse to robustness analysis, which is discussed in Section 3.5 and involves determining the set of plant polynomials $H(z)$ for which the above characteristic polynomials have stable roots.

3.4 Additive Noise Comparison of Open Loop and Closed Loop Performance

Assume that the uncertainties of the system are modeled as an additive zero mean noise on the plant output. That is:[5]

$$y(t+1) = \underline{h}^T \underline{u}(t) + w(t) \qquad (3.25)$$

The prediction model and the reference trajectory being as previously defined:

$$y_p(t+1) = \underline{\tilde{h}}^T u(t) \qquad \text{open loop} \qquad (3.26)$$

$$y_p(t+1) = y(t) + \tilde{h}^T(u(t) - u(t-1)) \qquad \text{closed loop} \qquad (3.27)$$

$$y_r(t+k) = \alpha^k y(t) + (1-\alpha^k)c \qquad k = 1,\ldots,T \qquad (3.28)$$

The cost function to minimize over a horizon of T is now, instead of J_T:

$$E[J_T] = E\left\{ \sum_{j=1}^{T} (y_p(t+j) - y_r(t+j))^2 \right\}$$

where E denotes the expectation operator.

Following the same line of argument as in Section 3.1, it is seen that when there is no constraint on inputs, the optimum input u*(t) is determined by the equation

$$y_p(t+1) = y_r(t+1) \qquad (3.29)$$

which translates into:

---

[5] Here, the output y(t+1) may be considered the measurement available to the controller. It includes the true plant output plus a measurement noise term.

$$\tilde{\underline{h}}^T u*(t) = \alpha y(t) + (1-\alpha)c \qquad \text{(open loop)} \qquad (3.30)$$

$$y(t) + \tilde{\underline{h}}^T(u*(t) - u*(t-1)) = \alpha y(t) + (1-\alpha)c \qquad \text{(closed loop)} \quad (3.31)$$

Similar to the deterministic case the optimum control sequence $u*(t)$ is determined sequentially from the above equations, in which the plant output $y(t)$ is observed. The difference with the deterministic case is that the present sequence of optimum control $u*(t)$ is nondeterministic, which results from the noisy nature of the plant output $y(t)$. The reader can easily verify that since the noise has zero mean the mean value $\bar{y}(t)$ of the plant output and the mean $\bar{u}*(t)$ of the optimum input sequence have a deterministic dynamic identical to the one governing the deterministic case as in Section 3.3. Thus, all the results of the previous section are valid for the calculation of the means $\bar{u}(t)$ and $\bar{y}(t)$. It remains to study the variance behavior, and in particular the variance of the controlled output $y(t)$. Here it becomes necessary to differentiate between the cases of white and colored noise.

3.4.1   White Noise

Let the noise be white, Gaussian, and independent from the input $u(t)$:

$$E[w(t)] = 0$$

$$E[w(t)w(\tau)] = \begin{cases} \sigma^2 & \text{for } t = \tau \\ 0 & \text{for } t \neq \tau \end{cases}$$

$$E[w(t)u(t)] = 0$$

Let us first assume that the identification of the impulse response is sufficiently good to allow the approximation $\tilde{\underline{h}}^T = \underline{h}^T$. Then the control

33

equation becomes:

$$y_p(t+1) = y(t+1) - w(t) = \alpha y(t) + (1-\alpha)c \quad \text{(open loop)} \quad (3.32)$$

$$y_p(t+1) = y(t) + h^T(u(t) - u(t-1)) = y(t-1) - w(t) + w(t-1)$$

$$= \alpha y(t) + (1-\alpha)c \quad \text{(closed loop)}$$

$$(3.33)$$

Denoting by $\sigma_0^2(t)$ and $\sigma_c^2(t)$ the variances of the open and closed loop plant output $y(t)$, one deduces:

$$\sigma_0^2(t+1) = \alpha^2\sigma_0^2(t) + \sigma^2 \quad \text{open loop} \quad (3.34)$$

$$\sigma_c^2(t+1) = \alpha^2\sigma_c^2(t) + 2\sigma^2(1-\alpha) \quad \text{closed loop} \quad (3.35)$$

which result in steady state variances of

$$\sigma_0^2(\infty) = \frac{\sigma^2}{1-\alpha^2} \quad \text{open loop} \quad (3.36)$$

$$\sigma_c^2(\infty) = \frac{2\sigma^2}{1+\alpha} \quad \text{closed loop} \quad (3.37)$$

It is seen that for the open loop prediction, the variance of the output decreases with α; that is, a fast reference model (small α) results in a smaller variance. But for the closed loop case, the variance of the output is a decreasing function of α; thus, a fast reference model results in a larger variance. Figure 9 displays that the breakeven point is for $\alpha = \frac{1}{2}$. Now, the practical implication of the above results is that when the plant output $y(t)$ is far from the desired value c, one can operate in

two consecutive phases; first use an open loop prediction with fast reference trajectory (α small) and then switch to a closed loop prediction control with slow reference trajectory. The latter is necessary, since as mentioned before, the open loop prediction yields a steady state bias.
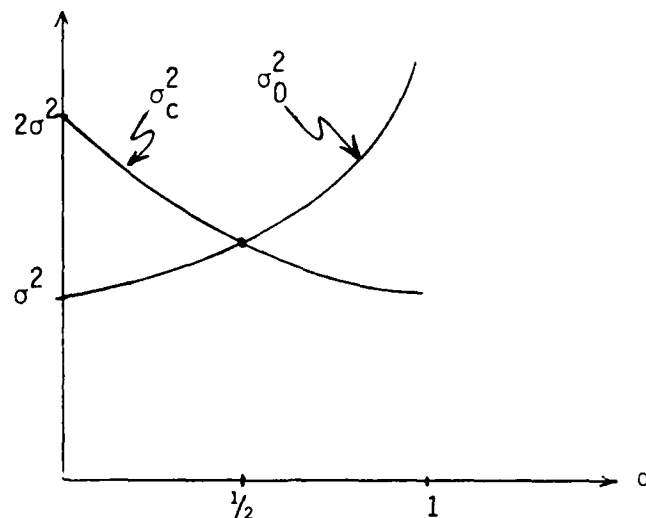
Figure 9. Closed loop and open loop variance.

Now assume that the identified impulse response $\tilde{\underline{h}}^T$ differs from the actual one $\underline{h}^T$. Then the evaluation of the output variance involves, as an intermediate stage, the computation of the covariance matrix:

$$U(\infty) = E[\underline{u}(\infty)\underline{u}^T(\infty)]$$

Or, equivalently, the evaluation of $E[u(t)u(t-i)]$ for $i = 0,\ldots,N$. It can be shown that this involves the solution of a system of $\frac{N(N+1)}{2}$ linear equations with $\frac{N(N+1)}{2}$ unknowns, this latter being elements of $U(\infty)$ (Anderson and Moore, 1979).

Let us consider the simple case of constant gain mismatch between the plant impulse response and the model one, i.e.,

35

$$\underline{h}^T = q\tilde{\underline{h}}^T \qquad\qquad q: \text{ gain}$$

Then the basic control equalities for open and closed loop prediction become:

$$y(t+1) = q\alpha y(t) + q(1-\alpha)c + w(t) \qquad \text{open loop} \qquad (3.38)$$

$$y(t+1) = [1 - (1-\alpha)q]y(t) + q(1-\alpha)c + w(t) - w(t-1) \qquad (3.39)$$
$$\text{closed loop}$$

The corresponding variance dynamics are:

$$\sigma_0^2(t+1) = (q\alpha)^2\sigma_0^2(t) + \sigma^2 \qquad \text{open loop} \qquad (3.40)$$

$$\sigma_c^2(t+1) = (1 - (1-\alpha)q)^2\sigma_c^2(t) + 2\sigma^2(1-\alpha)q \qquad \text{closed loop} \qquad (3.41)$$

whence the stability conditions for the convergence of σ_0^2 and σ_c^2 are:

$$|q\alpha| < 1 \qquad \text{open loop}$$
$$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (3.42)$$
$$|2 - (1-\alpha)q| < 1 \qquad \text{closed loop}$$

and the corresponding steady state variances are:

$$\sigma_0^2 = \frac{\sigma^2}{1 - (q\alpha)^2} \qquad \text{open loop}$$

$$\sigma_c^2 = \frac{2\sigma^2}{2 - (1-\alpha)q} \qquad \text{closed loop}$$

From (3.42), it is clear that the stability of both open loop and closed loop systems are guaranteed for:

$$0 < q < \min[\ \frac{2}{1-\alpha}\ ,\ \frac{1}{\alpha}\ ]$$

that is,

$$q < \frac{2}{1-\alpha} \qquad \text{for } -1 < \alpha < \tfrac{1}{3}$$

$$q < \frac{1}{\alpha} \qquad \text{for } \alpha > \tfrac{1}{3}$$

The plots of Figure 10 display the variances σ_0^2 and σ_c^2 as functions of the mismatch gain q in the above cases.
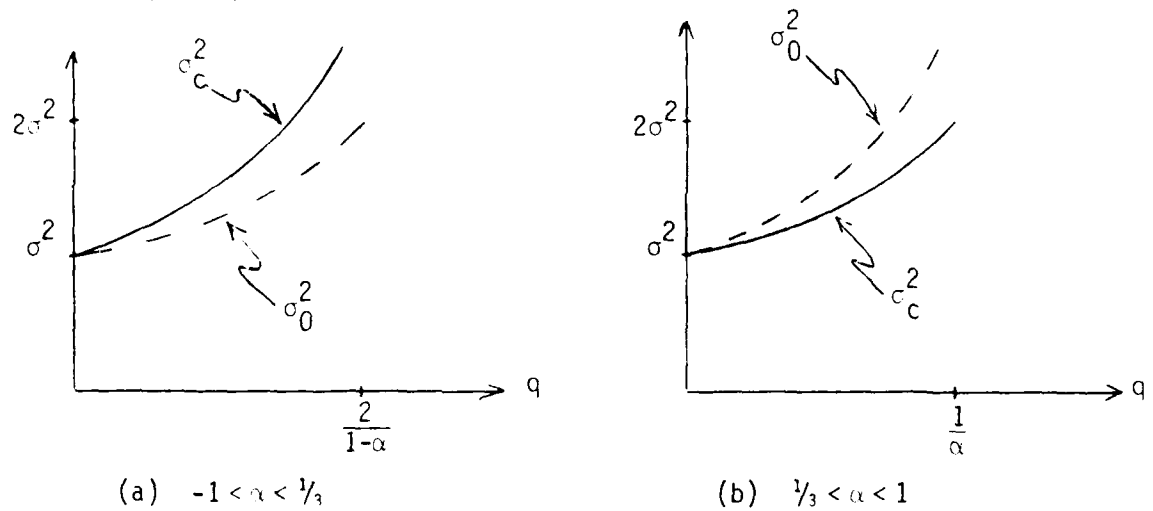
Figure 10. Noise variances with mismatch.

Let us first note that the parameter α is usually positive, since it is very unlikely and undesirable to select oscillatory reference trajectories. Plot (a) indicates that for a fast reference model ($0 < \alpha < \tfrac{1}{3}$), the open loop variance is always smaller than the closed loop one irrespective of the value of the gain q (provided that the system remains stable,

37

i.e., $q < 2/1-\alpha$). In contrast, plot (b) shows that for a slow reference model ($\frac{1}{3} < \alpha < 1$), the closed loop prediction always brings a smaller variance as compared to the open loop variance, and moreover for a gain factor of less than $1/\alpha$, the closed loop variance is less than $2\sigma^2$, i.e., twice the variance of the noise. Since in most practical applications one has to use a closed loop scheme to assure the unbiasedness, it is desirable for the reference trajectory to have a rate of growth α within the range of $[\frac{1}{3}, 1)$. This is not restrictive since in most cases, the reference trajectories rarely have rates of growth less than 0.6.

3.4.2   Colored Noise

If the noise is colored, then the relatively simple results of the previous subsection are no longer valid. The colored nature of the additive noise on the output is often due to the fact that the plant output $y(t)$ is observed through a filter, usually of low order. Moreover, any additive white process noise or input disturbance will reflect itself as output colored noise.

To see the type of behavior implied by the colored character of the noise, consider the simple case of a perfectly identified system ($\tilde{\underline{h}}^T = \underline{h}^T$) in the presence of a first order Markov process noise:

$$w(t+1) = \rho w(t) + e(t) \quad |\rho| < 1$$

$$e(t) : \text{zero mean white noise of variance } \nu^2$$

The stationary variance of the noise is then:

$$\sigma_w^2 = \frac{v^2}{1-\rho^2}$$

The systems equations are similar to the white noise case, but the recursive relations governing the variances are more complex.

For the open loop prediction scheme, it is seen without much difficulty that the variance recursion is:

$$\sigma_0^2(t+1) = \alpha^2 \sigma_0^2(t) + \sigma_w^2(t) + 2\alpha \ \mathrm{cov}(t)$$

where:

$$\mathrm{cov}(t,t) \triangleq E[(y(t) - \bar{y}(t)) \ w(t)]$$

and

$$\mathrm{cov}(t+1,t+1) = \alpha\rho \mathrm{cov}(t,t) + \rho\sigma_w^2(t)$$

Letting $t \to \infty$, one deduces the stationary variance of

$$\sigma_0^2 = \frac{1+\alpha\rho}{1-\alpha\rho} \ \frac{1}{1-\alpha^2} \ \sigma_w^2 = \frac{1+\alpha\rho}{1-\alpha\rho} \ \frac{1}{1-\alpha^2} \ \frac{v^2}{1-\rho^2}$$

As the plot of Figure 11 shows, for a positively correlated noise ($\rho > 0$), the qualitative behavior of the variance σ_0^2 with respect to α remains the same as in the white noise case; that is, an increase of α (slowing the reference model) results in a larger variance. But if the noise is negatively correlated ($\rho < 0$), then there is an optimum value of α, i.e.,

an optimum reference trajectory, associated with each ρ. And moreover, there is a range of values for α, resulting in an output variance smaller than the noise variance σ_w^2.
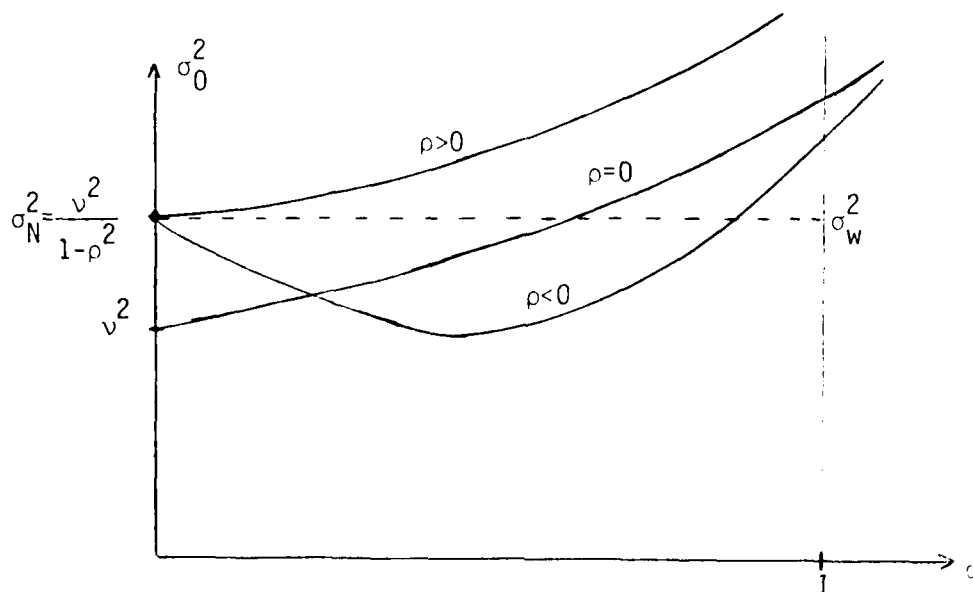
Figure 11. Open loop variance, colored noise.

For the closed loop prediction, from the equation (3.33), one deduces the following recursions for the variance:

$$\sigma_C^2(t+1) = \alpha^2 \sigma_C^2(t) + (1-2\rho)\sigma_w^2(t) + \sigma_w^2(t-1) + 2\alpha\,\mathrm{cov}(t,t) - 2\alpha\,\mathrm{cov}(t,t-1)$$

where

$$\mathrm{cov}\,(t,t) \triangleq E[(y(t) - \bar{y}(t))w(t)]$$
$$\mathrm{cov}(t,t-1) \triangleq E[(y(t) - \bar{y}(t))w(t-1)]$$

and:

40

$$\text{cov}(t+1,t+1) = \alpha\rho\text{cov}(t,t) + \rho\sigma_w^2(t) - \rho^2\sigma_w^2(t-1)$$

$$\text{cov}(t+1,t) = \alpha\text{cov}(t,t) + \sigma_w^2(t) - \rho\sigma_w^2(t-1)$$

Letting $t \to \infty$, the expression of the stationary variance is obtained as:

$$\sigma_c^2 = \frac{2}{1+\alpha}\ \frac{1-\rho}{1-\rho\alpha}\ \sigma_w^2 = \frac{2}{1+\alpha}\ \frac{1-\rho}{1-\rho\alpha}\ \frac{\nu^2}{1-\rho^2}$$

As the plots of Figure 12 show, for a negative correlation factor ρ, one has to slow down the reference trajectory (increase α) to decrease the variance. For positive ρ, three situations arise depending on the value of α. For $\rho \leq \frac{1}{3}$, the variance is still a decreasing function of α. For $\frac{1}{3} < \rho < \frac{1}{2}$, the variance reaches its minimum for $\alpha = \frac{1-\rho}{2\rho}$, and moreover, it is always bounded by its value at $\alpha = 0$, i.e., $2\nu^2/1+\rho$. For $\frac{1}{2} < \rho < 1$, the situation is reversed in the sense that the variance is bounded by its value at $\alpha = 1$, that is, by $\nu^2/1-\rho^2$. In these two latter cases, slowing down the reference model does not necessarily lead to a smaller output variance since the minimum is achieved for some $\alpha < 1$.
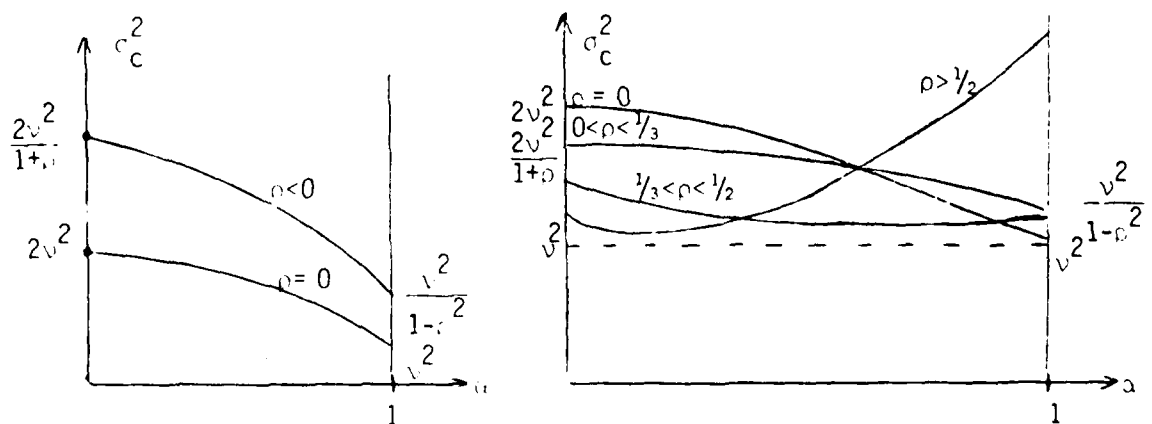
Figure 12. Closed loop variances, colored noise.

41

Figure 13 shows the ratio σ_c^2/σ_0^2 for different values of α and ρ. It is seen that the breakeven value of α, that is, the value for which the closed loop and the open loop variances are equal, moves toward the left as the noise correlation factor increases. Therefore, as the noise becomes more and more positively correlated, the range of value of α corresponding to a smaller closed loop variance becomes larger.
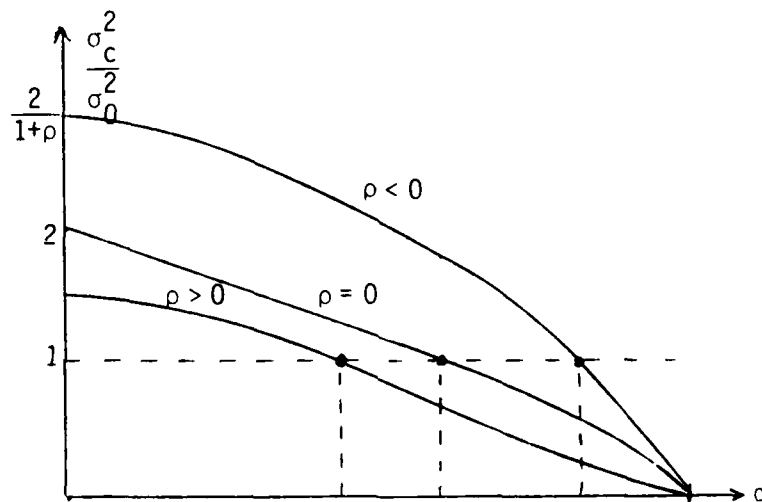
Figure 13. Closed loop - open loop variance ratio.

The above colored noise analysis shows that it is often advantageous to perform an output filtering, particularly when the output noise level is large. A proper choice of an output filter decreases the output variance while allowing the selection of a faster reference trajectory.

3.5 Robustness

The purpose of this section is to formulate and study the robustness problems associated with the control procedures described in the

previous sections. The control design is said to be robust if the plant output $y(t)$ converges to its ultimate desired value c, under a range of plant and behavior changes caused, for instance, by failure of some of its components or by large parameter deviations. The reader should also see Appendix B for an application of the ideas of this section to a simple missile example.

With regard to the closed loop prediction control mentioned previously, the robustness problem is posed in terms of the stability of the input sequence $u(t)$ and the plant output $y(t)$. This is a consequence, as mentioned in Section 3.3, of the fact that the stability of the closed loop system guarantees the unbiased convergence of the output to the desired set point c. Therefore, as long as the structural changes of the plant, which amount to altering the impulse response $\underline{h}^T$, are such that

$$(z-1)\tilde{H}(z) + (1-\alpha)H(z) \tag{3.43}$$

has all its roots within the unit circle, the control is robust.

Let us represent the relationship between the assumed impulse response $\underline{\tilde{h}}^T$ and the plant response $\underline{h}^T$ by a diagonal matrix R:

$$\underline{h}^T = \underline{\tilde{h}}^T R \qquad \text{or} \qquad \tilde{h}_i = r_i h_i \qquad i = 0,\ldots,N-1 \tag{3.44}$$

Then the characteristic equation is written as:

$$(z-1) \sum_{i=0}^{N-1} \tilde{h}_i z^{-i} + (1-\alpha) \sum_{i=0}^{N-1} r_i \tilde{h}_i z^{-i} = 0 \tag{3.45}$$

Given the model impulse response $\underline{\tilde{h}}^T$, it is of great interest to determine

43

the range of the matrix R such that (3.45) is stable. To do so one has
to determine a subspace θ of $\mathbb{R}^N$, such that for

$$(r_0, r_1, \ldots, r_{N-1}) \in \theta$$

the polynomial (3.45) remains stable. The subspace θ of $\mathbb{R}^N$ determines
uniquely, through (3.44), another subspace θ of $\mathbb{R}^N$ for impulse responses
$\underline{h}$ corresponding to stable closed loop characteristic polynomials (3.45).
This latter subspace is called the domain of robustness:

$$\phi = \left\{ \underline{v} \in \mathbb{R}^N \mid (z-1) \sum_{i=0}^{N-1} \tilde{h}_i z^{-i} + (1-\alpha) \sum_{i=0}^{N-1} v_i z^{-i} \text{ is stable} \right\}$$

Obviously the determination of such a subspace of $\mathbb{R}^N$ requires complex search
algorithms, and often, from a practical point, an unrealistic number of
computations. But fortunately, in almost all practical problems, valuable
additional information both on the physical properties of the plant and
on the failure and perturbation history of its different components is
available. The appropriate use of this information reduces substa   ly
the size and dimension of the search space. For instance, if the robust-
ness against identification errors or small perturbations is considered,
then the coefficients r_i are of the form:

$$r_i = 1 + \varepsilon_i \qquad i = 0, \ldots, N-1$$

where ε_i are small and their upper bounds can be estimated from the particular
identification scheme in operation. If the robustness against failure of
some components--sensors or activators--is of concern, as in Ackermann (1979),

44

then only a few elements of the impulse response change drastically; that is, most of the coefficients r_i are equal to 1. In addition to this reduction of the search space, various approximation procedures can be used which result generally in more conservative estimates of the robustness domain ϕ but have the advantage of computational efficiency.

Consider the case where the mismatch between the plant impulse response and the model is limited to pure gain, that is:

$$R = qI \qquad \text{or} \qquad r_i = q, \quad i = 0,\ldots,N-1$$

The characteristic polynomial of the closed loop prediction scheme becomes:

$$(z-1)\tilde{H}(z) + (1-\alpha)q\tilde{H}(z)$$

Since the model $\tilde{H}(z)$ is stable, the stability--i.e., robustness--condition becomes:

$$|1 - q(1-\alpha)| < 1$$

that is,

$$0 < q < \frac{2}{1-\alpha}$$

However, in practice one would limit the range of admissible gain q to $(0, \frac{1}{1-\alpha})$, since the range $(\frac{1}{1-\alpha}, \frac{2}{1-\alpha})$ results in an oscillatory response for y(t), which is not desirable. A similar analysis can be carried out to determine the phase margin for the controller.

45

3.6 Robustness Performance Index

So far we have been concerned with robustness in an absolute sense, that is, from the viewpoint of asymptotic convergence of the plant output to its desired value. Such a viewpoint does not provide an assessment of the relative robustness performance of different control schemes under various conditions. In particular, the effects of perturbations and components failure on the speed of convergence are not apparent at all, and clearly this is very important for industrial applications.

In light of the above considerations, it becomes necessary to define some type of "measure" for the performance of the control scheme with regard to robustness (Mehra et al., 1979). Here we propose two possible performance indices. First consider:

$$\rho' = \frac{D(y(t),c|\tilde{\underline{h}}^T = \underline{h}^T,\alpha)}{\underset{R\epsilon\theta}{\text{Sup}}\ D(y(t),c|\tilde{\underline{h}}^T R = \underline{h}^T,\alpha)} \tag{3.46}$$

where D denotes the total weighted distance between the plant output $y(t)$ and the desired set point c:

$$D = \sum_{t=0}^{\infty} (y(t) - c)^2 w_t \qquad w_t > 0$$

A decreasing sequence of weight w_t would enhance the importance of an early convergence to c. The numerator of the index is the distance conditioned on the perfect knowledge of the plant ($\tilde{\underline{h}} = \underline{h}$) and the use of a reference trajectory with rate α. In the denominator, first the distance is computed

46

assuming that the discrepancy between the plant $\underline{h}$ and the model $\underline{\tilde{h}}$ can be represented by $\underline{h} = \underline{\tilde{h}}R$, and the reference trajectory is specified by α. Then the maximum of this distance caused by variation of R within the range θ is considered. Both numerator and denominator depend on the reference trajectory through the value of α. Obviously the value of the performance index is within the range $(0,1)$, and the performance improves with increasing ρ'.

It is possible, and desirable, to define a performance index independent from the particular reference trajectory. Such an index can be used in comparison of MAC with other control schemes, such as LQ. One can define such an index by:

$$\rho'' = \frac{\underset{\alpha \in \Lambda}{\text{Inf}} \ D(y(t),c,\alpha | \underline{\tilde{h}}^T = \underline{h})}{\underset{R \in \theta}{\text{Sup}} \ \underset{\alpha \in \Lambda}{\text{Inf}} \ D(y(t),c,\alpha | \underline{\tilde{h}}^T R = \underline{h}^T)}$$

where $\Lambda \subset (0,1)$ is the range of admissible reference trajectories. The main difference with the previous case is that the controller is free to choose the best reference trajectory in the sense of minimizing the distance between the plant output and the desired path c. Here the numerator denotes the minimum distance achieved by a proper choice of α, under the condition of perfect identifiability. In the denominator, for each choice of R, the minimum distance is achieved by a proper choice of α; then the supremum is taken when R varies in the range θ.

The above performance indices are defined for deterministic models, but their generalization to the case corresponding to additive output

noise is straightforward by replacing the distances D with their expec-
tations. As an example, consider the case of pure gain mismatch q and
additive white noise, then the indices can be easily computed to have
the following expressions:

$$\rho' = \inf_{q \in \theta} \frac{2 - (1-\alpha)q}{1+\alpha} \qquad \theta = (q_1, q_2) \subset [0, \frac{2}{1-\alpha}]$$

$$\rho'' = \frac{\displaystyle\inf_{\alpha \in \wedge} \frac{1}{1+\alpha}}{\displaystyle\sup_{q \in \theta} \inf_{\alpha \in \wedge} \frac{1}{2-(1-\alpha)q}} \qquad \begin{cases} \theta = (q_1, q_2) \subset [0, \frac{2}{1-\alpha}) \\ \wedge = (\alpha_1, \alpha_2) \subset (0,1) \end{cases}$$

To see the behavior of ρ with respect to α, consider:

$$\frac{d\rho'}{d\alpha} = \frac{d}{d\alpha}\left[\frac{2-(1-\alpha)q_2}{1+\alpha}\right] = \frac{2(q_2-1)}{(1+\alpha)^2}$$

For $q_2 > 1$, that is, if the maximum possible mismatch gain is larger than
unity (which is usually the case), then slowing down the reference trajec-
tory (decreasing α) improves the robustness of the system.

The expression of ρ'' involves the maximum value of the gain and also
the maximum value of α which corresponds to the slowest admissible ref-
erence trajectory:

$$\rho'' = \frac{2 - (1-\alpha_2)q_2}{1+\alpha_2}$$

which decreases linearly when q_2 varies from 1 to $\frac{1}{1-\alpha_2}$.

It is interesting to note that the above results on robustness en-
hancement via slowing down the reference trajectory have been confirmed
in practical applications.

3.7 Amplitude Constraints on Inputs

In most applications, the input u(t) is not free of constraints. In general there are both amplitude and rate constraints imposed by technical and cost considerations. Under such constraints, the results of Section 3.2 must be revised; in particular, the minimization of the distance J_T over a horizon of length T (T > 1) does not necessarily yield the same optimal input u*(t) as the minimization of J_1 (T = 1).

It can be shown that, under rather weak conditions, the stability of the system free of input constraints implies the stability of the input constrained system. In fact, Praly (1979) (see Appendix A) has proved a more general result pertaining to nonlinearity in the recursive control equation determining u(t). Let us describe briefly his result, a complete proof of which involves mathematical technicalities and can be found in Appendix A. When there are no constraints, the recursive equation generating the inputs u(t) is linear, of the type:

$$u^*(t+1) = \frac{1}{h_0} \left[- \sum_{j=1}^{N-1} h_j u^*(t-j) + \alpha y(t-1) + (1-\alpha)c \right] \qquad (3.47)$$

Assume that the above recursion is stable and denote its steady state solution by $\bar{u}^*$. Now consider a nonlinear time varying function $f_t(\cdot)$ and assume that instead of (3.47) the recursion governing the input sequence is:

$$u^*(t+1) = f_t \left[\frac{1}{h_0} \left(- \sum_{j=0}^{N-1} h_j u^*(t-j) + \alpha y(t-1) + (1-\alpha)c \right) \right] \qquad (3.48)$$

49

Then in order for the stability of (3.47) to imply the stability of (3.48) it is necessary that at every time t, and for any v:

$$|f_t(\bar{u}^* + v) - \bar{u}^*| < \frac{1-\alpha}{r}|v| \tag{3.49}$$

where r is the greatest modulus of the roots of the characteristic equation corresponding to (3.47).

In the case of constant amplitude bounds on the input $u^*(t)$, the functions $f_t(\cdot)$ are time independent and involve saturation type of non-linearities:

$$f(v) \triangleq \begin{cases} M & \text{for } v \geq M \\ v & \text{for } m \leq v \leq M \\ m & \text{for } v \leq m \end{cases}$$

with $\bar{u}^* \in [m, M]$

It can be shown that such functions verify the inequality (3.49) when the recursion (3.47) is stable. Therefore, the nonlinearities introduced by amplitude constraints do not make the system unstable.

The above result is important since it reduces the absolute stability analysis of a MAC driven system with amplitude constraints to the one of the unconstrained system, which is relatively simple, thanks to its linearity. It should be noted that the necessary condition (3.49) is relatively general. A rather large class of nonlinear time varying functions verify such an inequality. These functions may reflect different types of constraints (such as time varying rate and amplitude constraints), and therefore the inequality (3.49) can be used to relate the stability analysis

50

of a system with more general types of constraints to that of the un-
constrained system.

3.8 Summary

A mathematical framework for the analysis of an idealized MAC
scheme has been presented. The main operations performed by MAC and the
corresponding components have been described. For single input-single
output systems, specific results pertaining to the stability and robust-
ness have been derived. An example of stability analysis of a MAC
control system for the single input-single output case is presented in
Appendix A.

SECTION IV

MAC FOR NONMINIMUM PHASE AND TIME DELAY SYSTEMS

Overview

In this section, the control of time delay and nonminimum phase
systems is discussed within the framework of MAC. It is shown that for
the control of such systems, MAC offers a relatively wide range of solu-
tions with various levels of performance. The implementations of these
solutions are fairly simple and often involve minor alteration of the
existing MAC software. However, from a theoretical point of view, there
is a major difference between minimum phase and nonminimum phase systems,
namely, the inverse of the former is stable whereas the inverse of the
latter is unstable. This necessitates the use of different models for
control computations and for prediction in MAC. The control model has the
property of possessing a stable inverse whereas the prediction model must
be as close as possible to the true model. In this section, an optimal
control model for use in MAC will be derived and its performance with other
commonly used techniques for nonminimum phase systems will be compared.

The organization of the present section is as follows. Section 4.1
presents a historical background and 4.2 presents a brief review of MAC,
describing its essential features for the single input-single output case.
Section 4.3 discusses the general nature of nonminimum phase systems in
the framework of MAC. It is also concerned with qualitative and conceptual
descriptions of the approaches of Sections 4.4 and 4.5. Section 4.4

52

describes the "least squares solution" for nonminimum phase systems. For the sake of clarity the detailed mathematical derivation of this latter solution is given in Appendix C. Section 4.5 presents a minimum phase solution based on removal of unstable poles. This is basically a pole placement type of solution. Section 4.6 describes more heuristic approaches which are nonetheless of great use and importance for applications. These heuristic approaches are basically trial and error types of solutions involving both cost coefficients and prediction interval adjustments.

4.1 Background

Control theory, from its very outset, has paid particular attention to the control of nonminimum phase systems. This interest has been motivated by the relatively frequent occurrence of nonminimum phase properties in various engineering control problems, particularly with multivariate systems.

All the available optimum control schemes--i.e., Linear Quadratic control (LQ), Minimum Variance control (MV), Self-Tuning (ST) and Model Adaptive control (MA)--have developed their own methodologies to tackle nonminimum phase systems (Astrom, 1970; Peterka, 1972; Astrom and Wittenmark, 1974; Ogata, 1970; Kwaakernak and Sivan, 1972). Despite the fact that these methodologies stem from different control philosophies, they have some common features. In particular, all of them lead in one way or another to suboptimal solutions with bounded inputs. The optimality of the solution is usually traded off for a meaningful realizable suboptimal solution (Astrom, 1970; Peterka, 1972).

The present section treats nonminimum phase systems in the framework
of MAC. As will be seen, the MAC methodologies for such systems offer a
set of possible solutions ranging from the "best suboptimal control,"
which involves an off-line solution of a Riccati equation, to a more
heuristic type of control involving pole placement and/or cost coefficient
adjustment. The general properties of these solutions are similar in
many respects to the ones derived from previously mentioned control schemes
(LQ, MV, ST, MA). In particular it is shown that there is always a trade-
off between the optimality of a solution and the boundedness of its cor-
responding input. The practical implementation of the MAC solutions for
nonminimum phase systems often needs only minor alterations of the existing
MAC software and hence may be realized at relatively low cost.

4.2 Brief Review of MAC

We recall from Section III that in MAC the true plant and its mathe-
matical model are represented by their impulse responses $\underline{h}^T$ and $\underline{\tilde{h}}^T$:

$$y(t+1) = \underline{h}^T \underline{u}(t) \qquad \text{Plant}$$

$$y_M(t+1) = \underline{\tilde{h}}^T \underline{u}(t) \qquad \text{Model}$$

(4.1)

with $\underline{h} \in \mathbb{R}^N$, $\underline{\tilde{h}} \in \mathbb{R}^N$, $\underline{u}(t) \in \mathbb{R}^N$. The vector $\underline{u}(t)$ represents the past N inputs
$u(t),\ldots,u(t-N+1)$.

Based on this mathematical model and on the observed output of the
plant, a linear prediction scheme is conceived:

$$y_p(t+1) = y_M(t+1) + [y(t) - y_M(t)]$$

$$\qquad (4.2)$$

$$= y(t) + \tilde{\underline{h}}^T[\underline{u}(t) - \underline{u}(t-1)]$$

The controller also generates a reference trajectory y_r, which is initialized on the observed output of the plant:

$$y_r(t+k) = \alpha^k y(t) + (1-\alpha^k)c; \qquad |\alpha| < 1$$

where c is the ultimate desired value to which the plant output must converge.

The control strategy of MAC is to force the predicted output y_p to follow, as closely as possible, the reference trajectory $y_r(t+k)$ over T future periods ($k = 1, \ldots, T$). Hence the controller must determine a set of T future inputs $u(t), \ldots, u(t+T-1)$ such that the weighted distance:

$$J_T = \sum_{i=1}^{T} [y_p(t+i) - y_r(t+i)]^2 w_i; \qquad w_i \geq 0 \qquad (4.3)$$

is minimized. It was shown in Section III that for a given $\underline{h}$, $\tilde{\underline{h}}$, α and c there exists a unique sequence of inputs such that the above minimum is zero, i.e., the predicted output y_p follows exactly the reference trajectory. In particular, if $\tilde{\underline{h}} = \underline{h}$, the corresponding optimum input sequence forces the plant output $y(t)$ to follow exactly the reference trajectory y_r. Therefore, if $\underline{h}$ is known, the best choice for $\tilde{\underline{h}}$, in the sense of minimizing the distance between the plant output and the reference trajectory, is $\underline{h}$.

For a one step minimization ($T = 1$), the optimum input $u^*(t)$, minimizing J_1, is determined by:

$$y_p(t+1) = y_r(t+1)$$

$$\text{or} \qquad y(t) + \tilde{h}^T[u(t) - u(t-1)] = \alpha y(t) + (1-\alpha)c \qquad (4.4)$$

In terms of z-transforms, denoting by $Y(z)$, $U(z)$, $C(z)$, $H(z)$ and $\tilde{H}(z)$ the transforms of the plant output $y(t)$, the plant input $u(t)$, the setpoint c and the impulse responses $\underline{h}$ and $\underline{\tilde{h}}$, one deduces:

$$Y(z) = \frac{H(z)}{(1-\alpha)H(z) + (z-1)\tilde{H}(z)} (1-\alpha)C(z) \qquad (4.5)$$

$$U(z) = \frac{z}{(1-\alpha)H(z) + (z-1)\tilde{H}(z)} (1-\alpha)C(z) \qquad (4.6)$$

These equations are equivalent to equations (3.21) and (3.22), with z^{-1} cleared from the transforms. This system is shown in Figure 14 (cf. Figure 8).
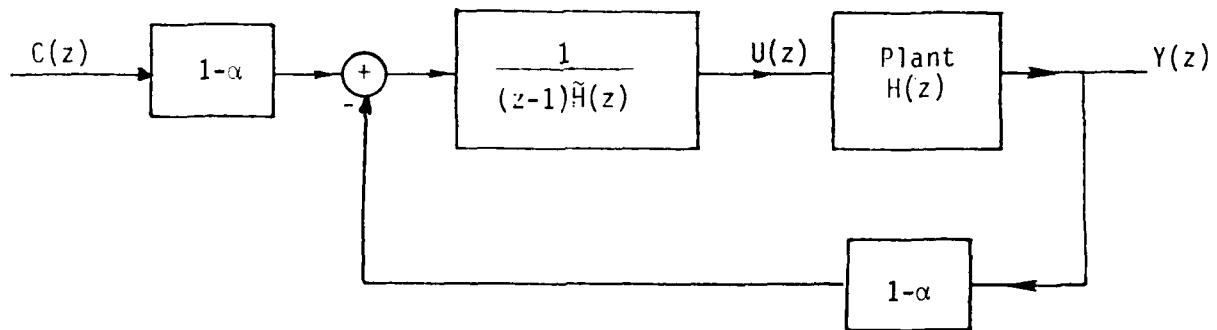

Figure 14. Idealized IDCOM block diagram.

4.3 Nonminimum-Phase System--Statement of the Problem

Let us assume that the impulse response $H(z)$ of the plant is perfectly known. Then, equating the model's response $\tilde{H}(z)$ with the plant's response, we get from equations (4.5) and (4.6),

$$Y(z) = \frac{H(z)}{(z-\alpha)\ \tilde{H}(z)}\ (1-\alpha)\ C(z) = \frac{1-\alpha}{z-\alpha}\ C(z) \tag{4.7}$$

$$U(z) = \frac{z}{(z-\alpha)\ \tilde{H}(z)}\ (1-\alpha)\ C(z) \tag{4.8}$$

Note that the plant output tracking transfer (i.e., $\frac{Y(z)}{C(z)}$) is of first order with unit gain and a single pole at α. This means that the plant output follows exactly the reference trajectory. The plant impulse response $H(z)$ does not intervene in the expression of $Y(z)$. But its *inverse* intervenes in the expression of the plant input $U(z)$. The zeros of $H(z)$ become poles of the plant input tracking transfer (i.e., $\frac{U(z)}{C(z)}$).

If $H(z)$ has some of its zeros outside the unit circle, that is, if the system is nonminimum phase, it is clear from equation (4.8) that the plant input $u(t)$ becomes unbounded. However, the output $y(t)$ remains bounded and follows the reference path as long as *none* of its unstable poles (corresponding to the cancelled $H(z)$ of equation (4.7)) are excited. It should be noted that practical considerations such as physical constraints, cost, etc. will always impose an upper bound on the norm of $u(t)$. Once this bound is reached, the cancellation of $H(z)$ in equation (4.7) is no longer perfect and the output $y(t)$ starts to diverge from the reference path. Figure 15 displays such a situation.

57

When H(z) is not known precisely, which is usually the case, the situation is worse in the sense that both y(t) and u(t) diverge. This is seen from the expression for Y(z) and U(z) in equations (4.5) and (4.6). The denominator $[(1-\alpha)H(z) + (z-1)\tilde{H}(z)]$ has its zeros close to the zeros of $(z-\alpha)H(z)$ (since $\tilde{H}(z)$ is close to H(z)). Therefore, Y(z) is unstable and divergent. (For further discussion of nonminimum phase systems, see Astrom, 1970; Peterka, 1972; Ogata, 1970; Astrom and Wittenmark, 1974.)

From the above analysis we deduce that unlike the minimum phase case where the natural choice of $\tilde{\underline{h}}^T$ (or equivalently $\tilde{H}(z)$) is a vector identical or close to $\underline{h}^T$, for the nonminimum phase case this choice of $\tilde{\underline{h}}^T$ does not lead to a stable and realizable control strategy. In fact, in order to have a stable control strategy which leads y(t) to its desired final value c, it is necessary to choose an $\tilde{H}(z)$ such that the polynomial $[H(z)(1-\alpha) + (z-\alpha)H(z)]$ has all its roots within the unit circle. This results in a stable input sequence u(t), which tends toward a steady state causing the term $\tilde{\underline{h}}^T[\underline{u}(t) - \underline{u}(t-1)]$ of equation (4.4) to converge to zero. Then as $t \to \infty$, $\lim y_p(t+1) = \lim y(t) = \lim y_r(t+1) = c$. Therefore, a proper choice of $\tilde{H}(z)$ (i.e., $\tilde{\underline{h}}^T$) will ensure the ultimate convergence of the output y(t) to the desired value c while keeping the input sequence bounded. However, as one might expect, the ultimate convergence of y(t) to the desired value c is not done at free cost; since $\tilde{\underline{h}}^T \neq \underline{h}^T$, the transitional behavior of y(t) is different from that of $y_p(t)$ (and hence from the reference model $y_r(t)$ which always equals $y_p(t)$ under the present control strategy). It is worth noting that by choosing $\tilde{\underline{h}}^T$ equal to $\underline{h}^T$,

the inverse situation occurs; that is, the output $y(t)$ follows perfectly
$y_p(t)$ (and hence the reference model y_r) during the transitional phase,
but diverges from it as t increases (Figure 15).

Obviously, there are an infinite number of vectors $\underline{\tilde{h}}^T$ resulting
in a stable characteristic polynomial $[H(z)(1-\alpha) + (z-1)\tilde{H}(z)]$. Among
all the admissible $\tilde{h}^T$ one may choose those which satisfy some secondary
criteria. A natural criterion is, for example, the minimization of the
total distance between the actual output $y(t)$ and the reference model
$y_r(t)$:

$$\min \sum_{j=1}^{\infty} [y(j) - y_r(j)]^2$$

Other criteria may involve direct pole placement for the characteristic
polynomial $[H(z)(1-\alpha) + \tilde{H}(z)(z-1)]$, or equivalently the design of the
dynamic of the error:

$$\varepsilon(t) = y_r(t) - y(t)$$

Sections 4.4 and 4.5 discuss the above cases.

Before ending this section we note that the use of an $\underline{\tilde{h}}^T$ in equation
(4.7) which is consciously different from $\underline{h}^T$ implies that the mathematical
model y_M is no longer attempting to "represent" the system. Therefore,
instead of y_M we introduce a new variable $\varepsilon(t)$ which measures the error
between the plant output $y(t+1)$ and the reference output $y_r(t+1)$:
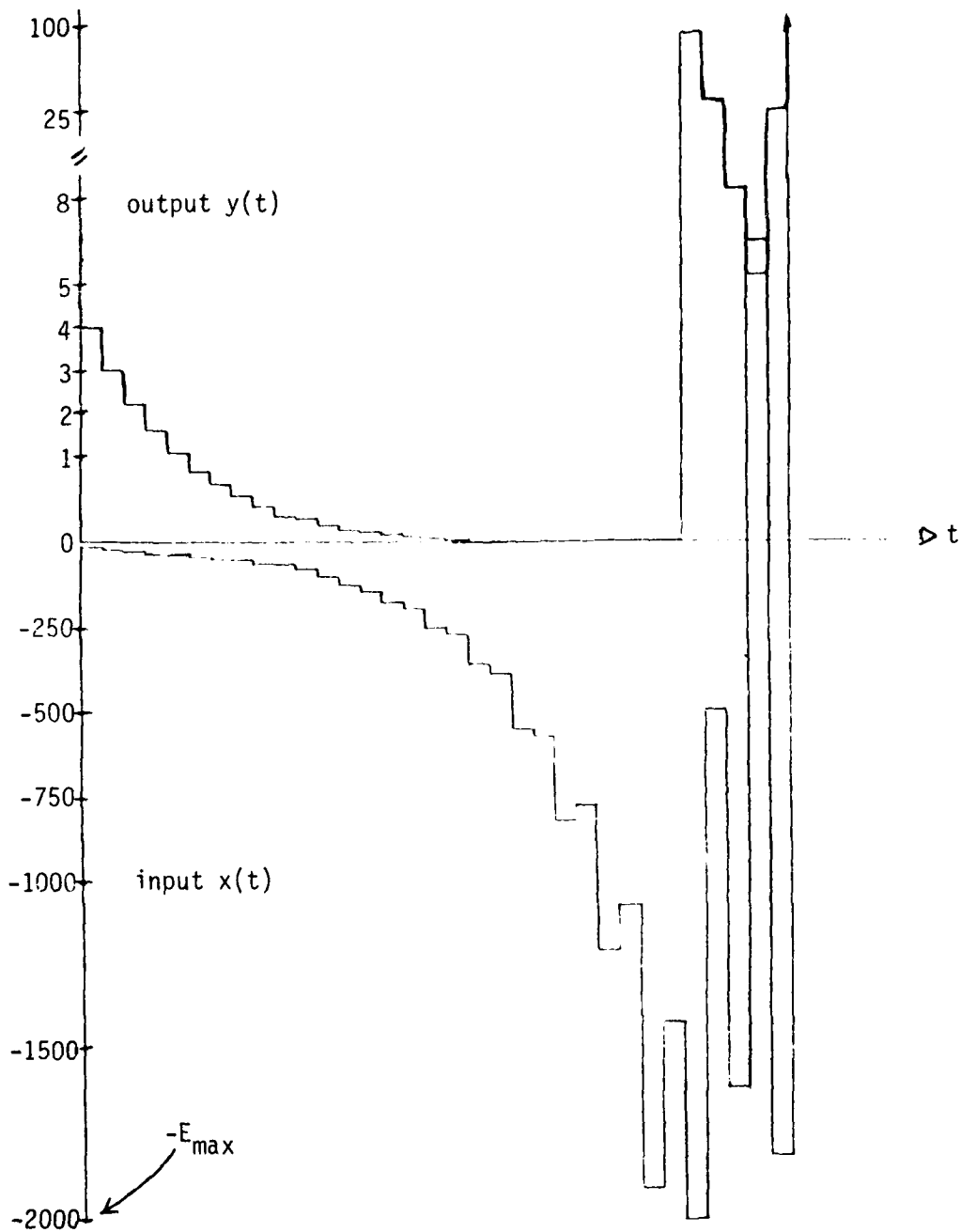
$h^T = [1., -.9, -1.26, 1.8060, -1.1092, .2856]$; $c = 0$, initial condition
$y(0) = 5$, $\alpha = .8$ and the bound on the value of the amplitude of the input
$E_{max} = 2000$. $\Delta T = 1$.

Figure 5. Simulation of the nonminimum phase system.

60

$$\varepsilon(t+1) = y_r(t+1) - y(t+1)$$

$$= y_p(t+1) - y(t+1)$$

$$= [\underline{\tilde{h}}^T - \underline{h}^T][\underline{u}(t) - \underline{u}(t-1)]$$

$$= \underline{d}[\underline{u}(t) - \underline{u}(t-1)]$$

with $\underline{d} = \underline{\tilde{h}}^T - \underline{h}^T$. As we can see, the vector $\underline{\tilde{h}}^T$ directly influences the dynamic of the error.

4.4 Least Square Solution

The present section deals with the determination of a vector $\underline{\tilde{h}}^T$ such that the total Euclidean distance between the actual output $y(t)$ and the reference model y_r is minimized. Minimize:

$$J = \tfrac{1}{2} \sum_{j=1}^{\infty} [y(j) - y_r(j)]^2 = \tfrac{1}{2} \sum_{j=1}^{\infty} \varepsilon^2(j) \qquad (4.9)$$

Hence the problem can be stated as follows: determine a vector $\underline{\tilde{h}}^T$ such that the characteristic equation $[(1-\alpha)H(z) + (z-1)\tilde{H}(z)]$ has all its roots within the unit circle and J is a minimum.

To solve the above problem, as one may expect, we will use the jargon of linear quadratic control. But first we have to formulate the problem in a framework suitable for a linear quadratic approach. For the sake of simplicity, and without loss of generality, we may assume that $c = 0$, i.e., the desired final value of the output is zero. Then the equations which determine the optimum sequence of input $u(t)$ (in the sense of matching $y_p(t+1)$ with $y_r(t+1)$) are, by equation (4.4):

$$y_r(t+1) = \alpha y(t) + \underline{h}^T(\underline{u}(t) - \underline{u}(t-1)) = y_p(t+1)$$

In terms of the sequence $u(t)$, we have:

$$\tilde{h}_1 u(t) = [\tilde{h}_1 - \tilde{h}_2 - (1-\alpha)h_1]u(t-1) + \ldots + [\tilde{h}_{N-1} - \tilde{h}_{N-2} - (1-\alpha)h_{N-1}]u(t-N)$$

$$+ [\tilde{h}_N - (1-\alpha)h_N]u(t-N-1) \tag{4.10}$$

The expression of the error $\varepsilon(t)$ in terms of $u(t)$ is derived as follows:

$$\varepsilon(t) = y(t) - y_r(t) = y(t) - \alpha y(t-1) = \underline{h}^T \underline{u}(t-1) - \alpha \underline{h}^T \underline{u}(t-2)$$

$$= [h_1, \; h_2 - \alpha h_1, \; \ldots, \; h_N - \alpha h_{N-1}, \; -\alpha h_N]\begin{bmatrix} u(t-1) \\ \vdots \\ u(t-N-1) \\ ---- \\ u(t-N-2) \end{bmatrix}$$

or equivalently,

$$\varepsilon(t) = \underline{g}^T\begin{bmatrix} u(t-1) \\ ---- \\ u(t-N-1) \end{bmatrix} \text{ with } \underline{g}^T = [h_1, \; h_2 - \alpha h_1, \; \ldots, \; h_N - \alpha h_{N-1}, \; -\alpha h_N] \tag{4.11}$$

Note that the dimension of $\underline{g}^T$ is $(N+1)$, when the dimension of $\underline{h}^T$ is N. Let us introduce the $(N+1)$-dimensional vector $\underline{\xi}(t)$ defined as:

$$\underline{\xi}^T(t) = [\underline{u}^T(t) \mid u(t-N-1)] = [u(t), u(t-1), \ldots, u(t-N), u(t-N-1)]$$

Then $\varepsilon(t)$ and J can be written as

$$\epsilon(t) = \underline{g}^T \underline{\xi}(t)$$

(4.12)

$$J = \lim_{t \to \infty} [\tfrac{1}{2}\underline{\xi}^T(t)\underline{q}\underline{q}^T\underline{\xi}(t) + \sum_{j=1}^{t-1} \xi^T(j)\underline{q}\underline{q}^T\underline{\xi}(j)]$$

Equation (4.10) can be written in terms of $\underline{\xi}(t)$ as:

$$\underline{\xi}(t) = U\underline{\xi}(t-1) + \underline{b}_1\{[\underline{L}^T \mid 0]\underline{\xi}(t-1)\}$$

(4.13)

with U, $\underline{b}_1$, and $\underline{L}^T$ defined as:

$$U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \updownarrow (N+1) \qquad b_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \updownarrow (N+1)$$

(4.14)

$$\underline{L}^T = \frac{1}{\tilde{h}_1}[\tilde{h}_1 + \tilde{h}_2 - (1-\alpha)h_1, \ldots, \tilde{h}_{N-1} + \tilde{h}_N - (1-\alpha)h_{N-1}, \tilde{h}_N - (1-\alpha)h_N]$$

$$\underline{L} \in \mathbb{R}^N$$

Equation (4.13) is a state equation with a classical state feedback control law. It is clear that the determination of the optimum $\tilde{\underline{h}}^T$ which minimizes J is equivalent to the determination of a linear feedback gain $[\underline{L}^T | 0]$ of equation (4.13) minimizing J. At this point there is sufficient motivation to formulate the problem in the following way.

Find the optimum input sequence $u^*(t)$ which minimizes

$$J = \lim_{k \to 0} \tfrac{1}{2}[\xi^T(k)\underline{q}\underline{q}^T\xi(k) + \sum_{j=1}^{k-1} \xi^T(j)\underline{q}\underline{q}^T\xi(j)]$$

(4.15a)

63

where the state equation is:

$$\xi(t) = U\xi(t-1) + \underline{b}_1 u^*(t-1) \tag{4.15b}$$

It is easy to check that the above system is controllable. The optimum control $u^*(t)$ is a linear function of the state vector $\xi(t)$: $u^*(t) = \hat{\underline{L}}^T \xi(t)$ (Sage and White, 1977; Murata, 1977). It remains to show that $\hat{\underline{L}}^T$ has the property of having its last element $\hat{L}_{N+1}$ equal to zero. The following lemma establishes that result.

Lemma: The optimal input sequence $u^*(t)$ for equation (4.15) does not depend on the last element of the state vector $\xi(t)$.

Proof: By linear quadratic control theory, we know that $u^*(t)$ has the following expression in terms of $\xi(t)$:

$$u^*(t) = -\{[R + \underline{b}_1^T P \underline{b}_1]^{-1} \underline{b}_1^T PU\}\xi(t) = \hat{\underline{L}}\xi(t)$$

where R is the cost matrix of the input $u(t)$ (here $R = 0$ since J depends only on the state vector and P is the steady state solution of a matrix Riccati equation). Then,

$$b_1^T P b_1 = P_{11}$$

$$b_1^T PU = [P_{11}, P_{12}, \ldots, P_{1(N+1)}]U = [P_{12}, P_{13}, \ldots, P_{1(N+1)}, 0]$$

and hence

$$\hat{L} = \frac{-1}{P_{11}} [P_{12}, P_{13}, \ldots, P_{1(N+1)}, 0]$$

∎

Let us recapitulate the results so far. It has been shown that the determination of the best vector $\underset{\sim}{h}^T$ in the sense of minimizing the total Euclidean distance between the plant output and the reference trajectory while keeping the control sequence $u^*(t)$ bounded is equivalent to the determination of the gain matrix $\hat{L}$ of the linear quadratic control problem (4.15). The determination of $\hat{L}$ involves the solution of a Riccati equation of order N+1. Once $\hat{L}$ is known $\underset{\sim}{h}^T$ can be computed uniquely from the last equation of (4.14). It should be noted that $\underset{\sim}{h}^T$ depends uniquely on h^T and α (the rate of growth of the reference model). Since the determination of $\underset{\sim}{h}^T$ is done off-line, even for large values of N (of order 80) the vector $\underset{\sim}{h}^T$ can be computed without excessive cost. The linear quadratic control problem of (4.15) is further detailed in Appendix C. We end the present section by applying the above considerations to the example of the previous section (see Figure 16).

4.5 Direct Pole Placement

In addition to the above optimal choice in the sense of quadratic error minimization, there are an infinity of other possible choices of $\underset{\sim}{h}^T$ leading to stable control solutions. The vector $\underset{\sim}{h}^T$ is to be chosen such that the polynomial $[H(z)(1-\alpha) + (z-1)\tilde{H}(z)]$ has all its roots confined within the unit circle. Therefore, determining H(z) amounts to identifying the above characteristic polynomial with an <u>arbitrary</u> stable polynomial of order N+1.

However, going back to the original definition of $\underset{\sim}{h}^T$ and $\underset{\sim}{h}^T$, and noting that for minimum phase systems one equates $\underset{\sim}{h}^T$ with $\underset{\sim}{h}^T$, it becomes
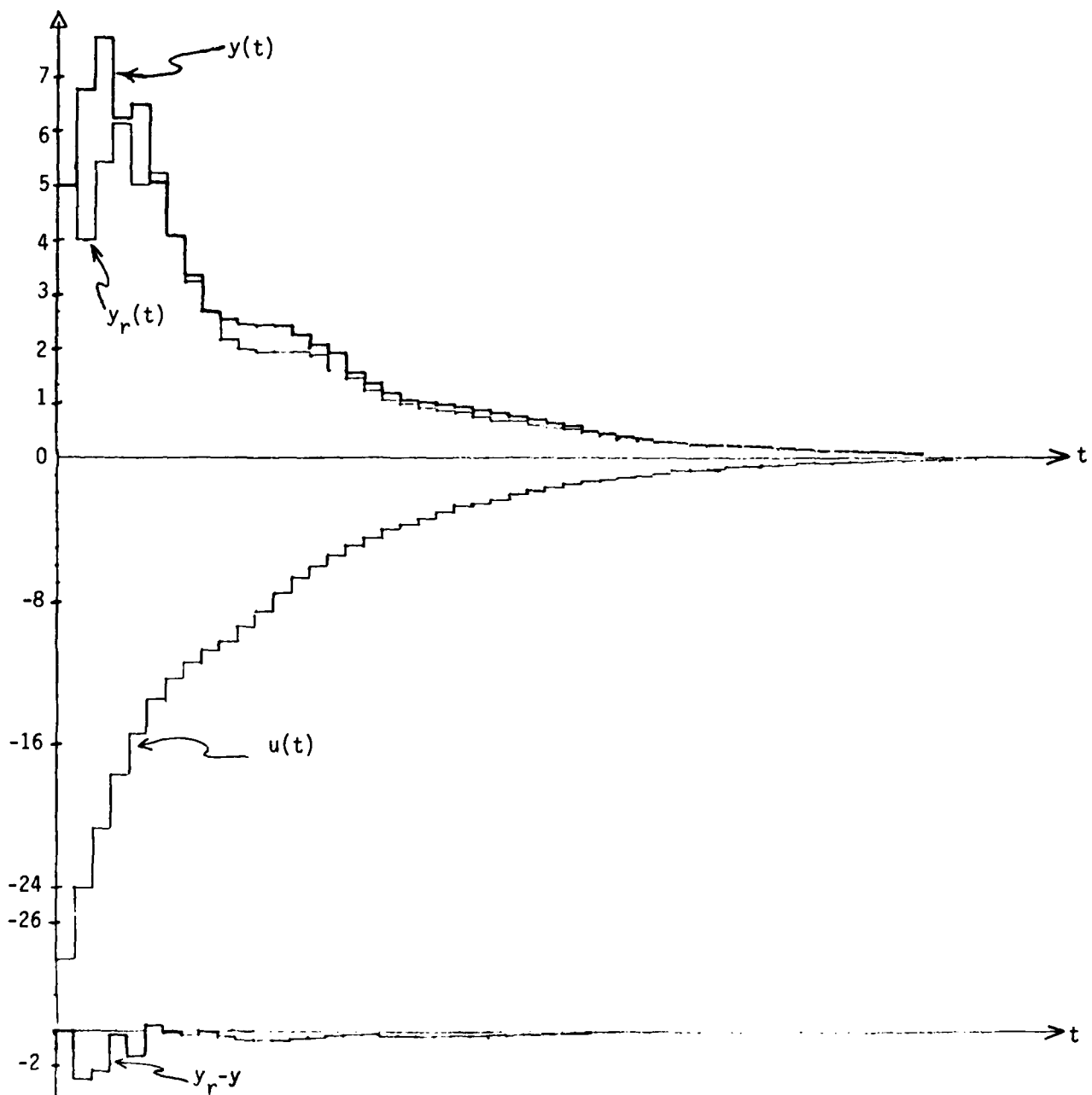
$\underline{h}^T = [1., -.9, -1.26, 1.806, -1.1092, .2856]$ with $\alpha = .8$ and $c = 0$. The optimal $\underline{\tilde{h}}^T$ is $[-.5754, .0546, .0446, .1266, -.2946, .0772]$.

Figure 16. LQ optimal control.

natural to conceive a design procedure for $\tilde{H}(z)$ which remains consistent with the minimum phase case. Loosely speaking, $\tilde{H}(z)$ should retain "as much as possible" of the dynamic properties of $H(z)$, while stabilizing the characteristic polynomial $[H(z)(1-\alpha) + (z-1)\tilde{H}(z)]$. One way of achieving that objective is to maintain the stable roots of $H(z)$ and to alter only the unstable ones. Let us write $H(z)$ as:

$$H(z) = B_{0s}(z)B_{0u}(z) \tag{4.16}$$

where $B_{0s}(z)$ contains all the stable roots of $H(z)$ and $B_{0u}(z)$ the remaining unstable ones. Then let

$$\tilde{H}(z) = B_{0s}(z)B_{Ms}(z) \qquad \begin{cases} \deg\ B_{0s} = N-\ell \\ \deg\ B_{Ms} = \ell \end{cases} \tag{4.17}$$

where $B_{0s}(z)$ has the same stable zeros as $H(z)$. The characteristic polynomial $[H(z)(1-\alpha) + (z-1)\tilde{H}(z)]$ becomes

$$B_{0s}(z)[B_{0u}(z)(1-\alpha) + (z-1)B_{Ms}(z)] \tag{4.18}$$

$B_{Ms}(z)$ must be chosen so that the polynomial

$$B_{0u}(z)(1-\alpha) + (z-1)B_{Ms}(z) \tag{4.19}$$

is stable. Once $B_{Ms}(z)$ is specified, the polynomial $\tilde{H}(z)$ and thus the vector $\tilde{\underline{h}}^T$ are determined. The resulting control sequence $u(t)$ will display the stable modes of $H(z)$ in addition to the one of $[B_{0u}(z)(1-\alpha) + (z-1)B_{Ms}(z)]$. choice for this latter polynomial is the one having $(\ell+1)$ zeros at

the origin:

$$B_{0u}(z)(1-\alpha) + (z-1)B_{Ms}(z) = C_1 z^{\ell+1}$$

(4.20)

or $\quad (1-z)B_{Ms}(z) = C_1 z^{\ell+1} - (1-\alpha)B_{0s}(z)$

with C_1 being a real constant.

The above choice of $B_{Ms}(z)$ will lead to a fast convergence of $y(t)$, although not a very smooth one. Other choices of (4.19) may lead to a smoother convergence of $y(t)$. In particular, $B_{Ms}(z)$ may be determined so that the polynomial (4.19) becomes identical to:

$$C_2 z^k (z^{\ell+1-k} - \Gamma)$$

with C_2 being a real constant, $0 \leqslant k \leqslant \ell+1$, and $|\Gamma| \ll 1$. That is, the $(\ell+1)$ unstable poles of the original system are replaced by k poles at the origin and $(\ell+1-k)$ poles of module $|\Gamma|^{1/\ell+1-k}$. Figures 17 and 18 display the application of the above considerations to the example of Sections 4.4 and 4.5.

Before ending this section, let us notice that: (i) the polynomial $B_{0u}(z)$ may also include stable poles which are close to the unit circle. By doing so, one is assured that "small" perturbations of $H(z)$ will not destabilize the controlled system.

(ii) The main advantage of the "direct pole placement" approach over the linear quadratic one lies in its greater flexibility as compared with the rigidity of the unique aspect of the LQ solution. This flexibility, which is the result of the relative freedom involved in the choice

68
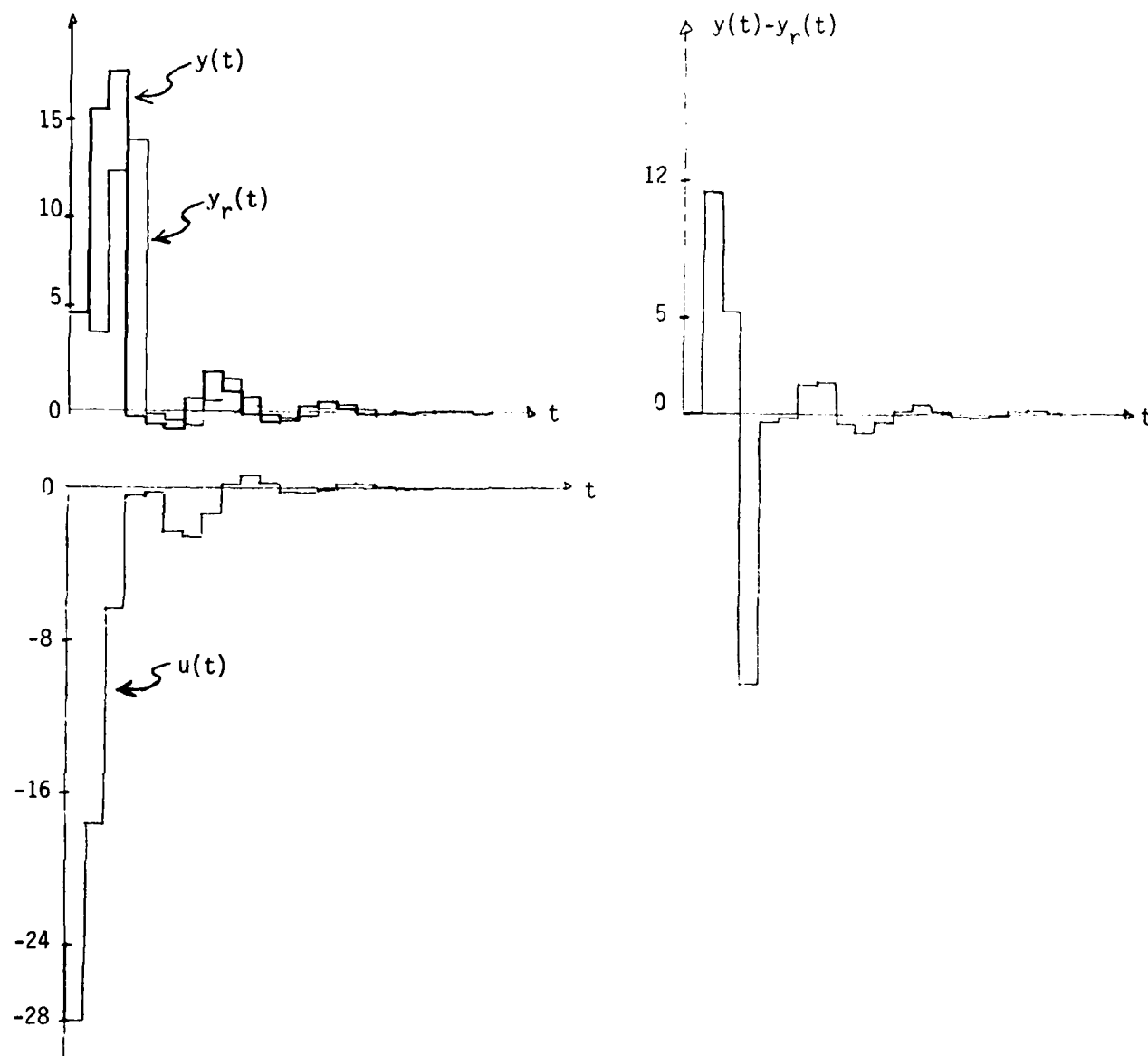
$h^T = [1., -.9, -1.26, 1.806, -1.1092, .2856]$, $\tilde{h}^T = [-.096, -1.904, -.07376, .1915, -.1685, .0605]$. Poles of the nonminimum phase system: $.3 \pm 5j$, $.5$, 1.2, 1.4. Poles of $[H(z)(1-\alpha) + (z-1)\tilde{H}(z)]$: $.3 \pm 5j$, $.5$, 0, 0, 0. $\alpha = .8$, $c = 0$.

Figure 17. Stabilization by pole placement.

$\underline{h}^T = [1., -.9, -1.26, 1.806, -1.1092, .2856]$; $\tilde{\underline{h}}^T = [-.128, -.1872, -.05968, .17639, -.16272, .06048]$. $\alpha = .8$, $c = 0$.
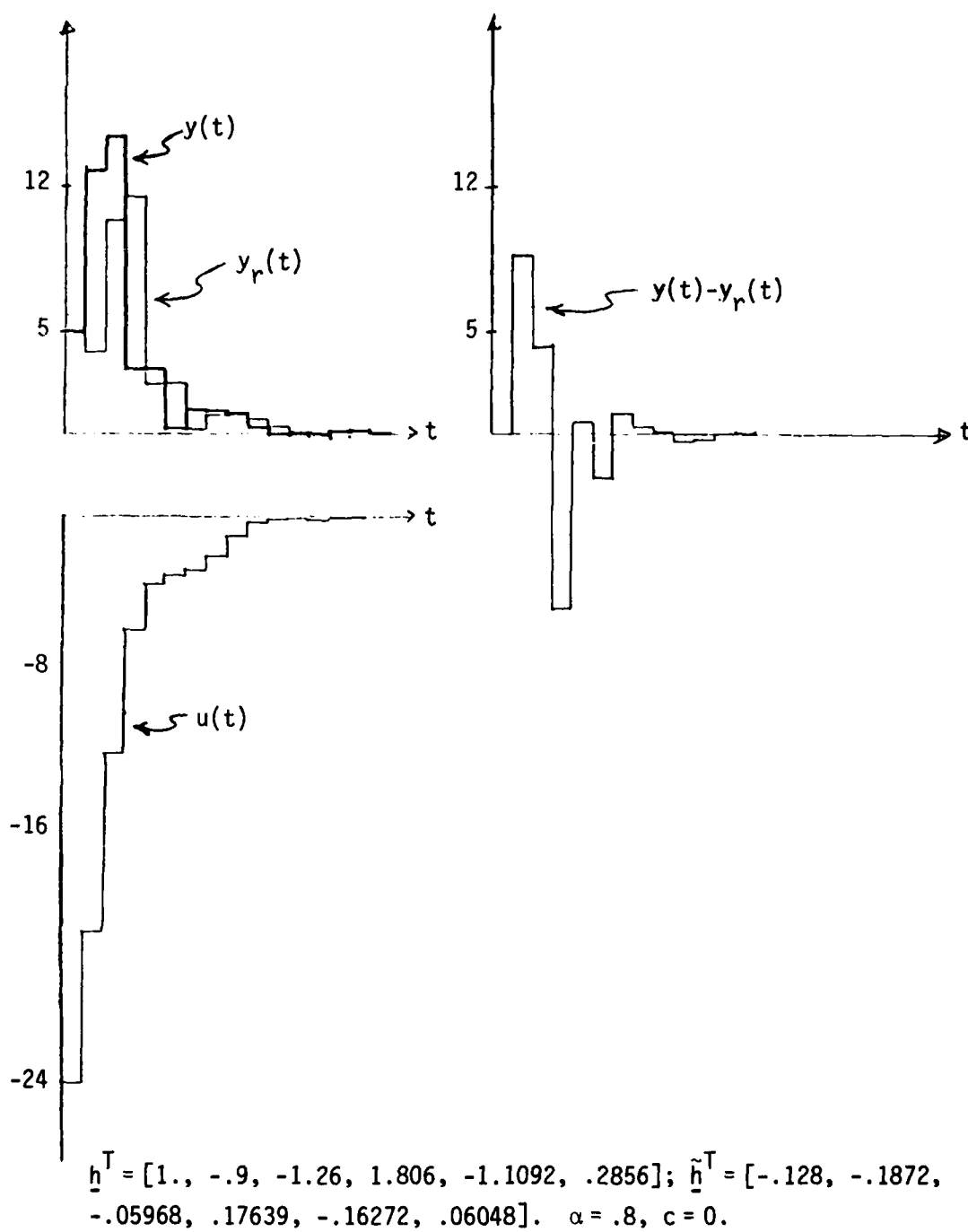
Figure 18. Stabilization by pole placement.

of $B_{Ms}(z)$, allows one to influence some important aspects of the converging output $y(t)$ such as its smoothness, overshoot and undesired oscillations. In many practical problems one is willing to sacrifice the "minimum quadratic error" nature of the linear quadratic solution for an improvement in the other more important features, such as those mentioned above.

4.6 Constant Input Over p Steps

In the present section we describe another somewhat more heuristic approach to the control of nonminimum phase systems. The present approach is different from the above in that it requires some alteration of the structure of MAC strategy. However, at the level of implementation these necessary alterations are done via minor changes in the existing MAC software.

Let us recall that in the regular MAC, at each period t, one looks for p future optimum inputs $\{u(t), u(t+1),..., u(t+p-1)\}$ such that the sum

$$\sum_{i=0}^{p-1} w_i |y_r(t+i-1) - \alpha^i y(t)|^2 = \sum_{i=0}^{p-1} w_i |y(t+i) + \tilde{h}(\underline{u}(t+i) - \underline{u}(t+i-1)) - \alpha^i y(t)|^2, \qquad w_i \geqslant 0$$

(4.21)

is minimized.

Now let us impose the restriction of constant control over the p future inputs, i.e.,

$$u(t) = u(t+1) = \ldots = u(t+p-1)$$

Then the problem is reduced to the determination of the single value u(t) such that (4.21) is minimized. It is clear that:

$$\min_{\{u(t),\ldots,u(t+p-1)\}} \sum_{i=0}^{p-1} w_i |y_p(t+i+1) - \alpha^i y(t)|^2$$

$$\leqslant \min_{\{u(t) = \ldots = u(t+p-1)\}} \sum_{i=0}^{p-1} w_i |y_p(t+i+1) - \alpha^i y(t)|^2$$

Moreover, in the minimization shown in the right-hand side of the above inequality, the optimum value of u(t) depends on the length p of the horizon of prediction, while in the left-hand side the optimum value of the input sequence $\{u(t), u(t+1),\ldots, u(t+p-1)\}$ is independent of p. Schematically the differences are shown in Figure 19.
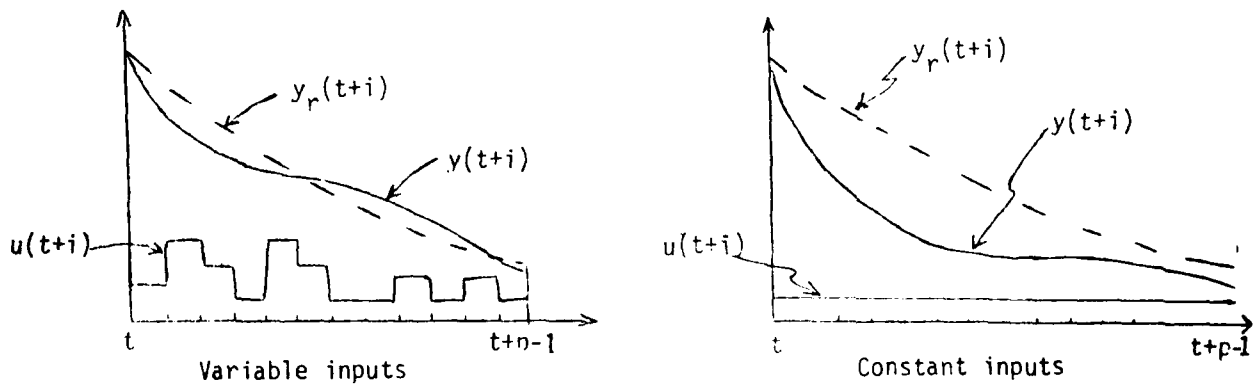

Figure 19. Effect of constant inputs on optimization.

Another important property is that the stability of the system is going to be influenced not only by the reference model (via the choice of α) but also by the p parameters w_i. It will be seen that proper choice of w_i's $(i = 0, \ldots, p-1)$ can stabilize a nonminimum phase system. Again, it is worth noting that the introduction of the horizon p and the parameters w_i $(i = 0, \ldots, p-1)$ allowing the stabilization is done at the cost of degrading the distance minimization between the reference model and the actual output during a transitional period.

In the following two subsections we treat the case of $p = 2$. The results are easily extendable to $p > 2$.

4.6.1   Two Step Ahead Prediction

Consider the particular case where $p = 2$ and $w_2 = 1$. That is, at period t, we are seeking to determine the inputs u(t) and u(t+1) such that the predicted $y_p(t+2)$ matches perfectly the desired reference $y_r(t+2)$:

$$\left.\begin{aligned} y_r(t+1) &= \alpha y(t) \text{ (we assume that } c = 0) \\ y_r(t+2) &= \alpha^2 y(t) \end{aligned}\right\} \text{ reference model} \qquad (4.22)$$

$$\left.\begin{aligned} y_p(t+1) &= y(t) + \tilde{\underline{h}}^T[\underline{u}(t) - \underline{u}(t-1)] \\ y_p(t+2) &= y(t) + \tilde{\underline{h}}^T[\underline{u}(t+1) - \underline{u}(t)] \end{aligned}\right\} \text{ prediction model} \qquad (4.23)$$

$$\left.\begin{aligned} y(t+1) &= \underline{h}^T \underline{u}(t) \\ y(t+2) &= \underline{h}^T \underline{u}(t+1) \end{aligned}\right\} \text{ plant} \qquad (4.24)$$

The equality condition becomes:

$$y_r(t+2) = y_p(t+2) \tag{4.25}$$

That is,

$$\tilde{h}_1 u(t+1) + \tilde{h}_2 u(t) = \{-(1-\alpha^2)\underline{h}^T + \tilde{\underline{h}}^T(I-U^2)\}\underline{u}(t-1)$$

where $\tilde{h}_1$, $\tilde{h}_2$ are the first two elements of $\tilde{\underline{h}}$, and U^2 is the two step ahead shift matrix. The additional assumption that $u(t) = u(t+1)$ (constant input over the prediction horizon) implies that:

$$u(t) = u(t+1) = \frac{1}{\tilde{h}_1 + \tilde{h}_2}\{-(1-\alpha^2)\underline{h}^T + \tilde{\underline{h}}^T(I-U^2)\}\underline{u}(t-1) \tag{4.26}$$

N can be assumed even (if it is not, we can always consider a new impulse response of length N+1 with $h_{N+1} = 0$). Therefore, $N = 2q$. Then let us define a new vector $\underline{v}$ of dimension N/2 such that:

$$\underline{v}(t) = u(t) = u(t+1)$$

$$\underline{v}(t+1) = u(t+2) = u(t+3)$$

$$\vdots$$

$$\underline{v}(t+q-1) = u(t+2q-2) = u(t+2q-1)$$

Thus equation (4.25) becomes:

$$
\begin{aligned}
(\tilde{h}_1 + \tilde{h}_2)v(t) = {}& [\tilde{h}_1 + \tilde{h}_2 - (1-\alpha^2)(h_1 + h_2)]v(t-1) \\
& + [\tilde{h}_3 + \tilde{h}_4 - (1-\alpha^2)(h_3 + h_4)]v(t-2) \\
& + \ldots \\
& + [\tilde{h}_{N-1} + \tilde{h}_N - (1-\alpha^2)(h_{N-1} + h_N)]v(t-q)
\end{aligned}
\tag{4.27}
$$

74

Define:

$$g = \begin{bmatrix} h_1 + h_2 \\ h_3 + h_4 \\ \vdots \\ h_{N-1} + h_N \end{bmatrix} \quad \text{and} \quad \tilde{g} = \begin{bmatrix} \tilde{h}_1 + \tilde{h}_2 \\ \tilde{h}_3 + \tilde{h}_4 \\ \vdots \\ \tilde{h}_{N-1} + \tilde{h}_N \end{bmatrix}$$

Then the above equations can be written as:

$$\tilde{g}_1 \nu(t) = [\tilde{g}^T(I-U) - (1-\alpha^2)g^T]\underline{\nu}(t-1)$$

$$= [\tilde{g}_1 - \tilde{g}_2 - (1-\alpha^2)g_1]\underline{\nu}(t-1) + \ldots + [\tilde{g}_q - (1-\alpha^2)g_q]\underline{\nu}(t-q) \tag{4.28}$$

where $\tilde{g}_i$ and g_i are components of $\tilde{g}$ and g, or:

$$g^T\underline{\nu}(t) = \alpha^2 g^T\underline{\nu}(t-1) + \tilde{g}^T[\underline{\nu}(t) - \underline{\nu}(t-1)] \tag{4.29}$$

The reader familiar with MAC formulation will easily see the similarities of equations (4.27) and (4.28) with equations (4.8) and (4.10) describing the one-step prediction scheme. Equations (4.27) and (4.26) can be deduced from (4.8) and (4.10) by replacing $\underline{h}$, $\tilde{\underline{h}}$ and α with g, $\tilde{g}$ and α^2, respectively. The difference is in the dimension of the system (N/2 instead of N). All the considerations of Sections 4.4 and 4.5 can be applied to the present system of dimension N/2. That is, if g^T is minimum phase, then one would choose $\tilde{g}^T = g^T$. Then equation (4.26) will yield a stable input sequence $\nu(t)$ and by equations (4.22) and (4.24), we will have

$$y(t+2) = y_r(t+2) = \alpha^2 y(t) \tag{4.30}$$

while

$$y(t+1) \neq y_r(t+1) = \alpha y(t)$$

Then by equation (4.30),

$$\lim_{t \to \infty} y(t+2) = \lim_{t \to \infty} y_r(t+2) = 0$$

and by equation (4.23),

$$\lim_{t \to \infty} y(t+1) = \lim_{t \to \infty} y(t) + \underline{\tilde{h}}^T \lim_{t \to \infty} [\underline{u}(t) - \underline{u}(t-1)] = \lim_{t \to \infty} y(t) + 0$$

Hence,

$$\lim_{t \to \infty} y(t+1) = \lim_{t \to \infty} y(t) = \lim_{t \to \infty} y_r(t) = 0$$

That is, although during a transitional phase the perfect matching of y_r and y occurs each two steps $(t+2, t+4, \ldots, t+2j)$, as t increases $y(t)$ converges to a desired value c (here $c = 0$). Figure 20 shows such a behavior for a 6-order system.

Now if the vector g^T corresponds to a nonminimum phase system, then one has to determine an N/2-dimensional vector $\tilde{g}^T$ stabilizing the dynamics of (4.28). Both methods of Sections 4.4 and 4.5 can be used. The clear advantage is that the dimension N has been reduced by half. It is easy to see that a p-steps ahead prediction scheme would reduce the dimension of the system to be stabilized to N/p instead of N.
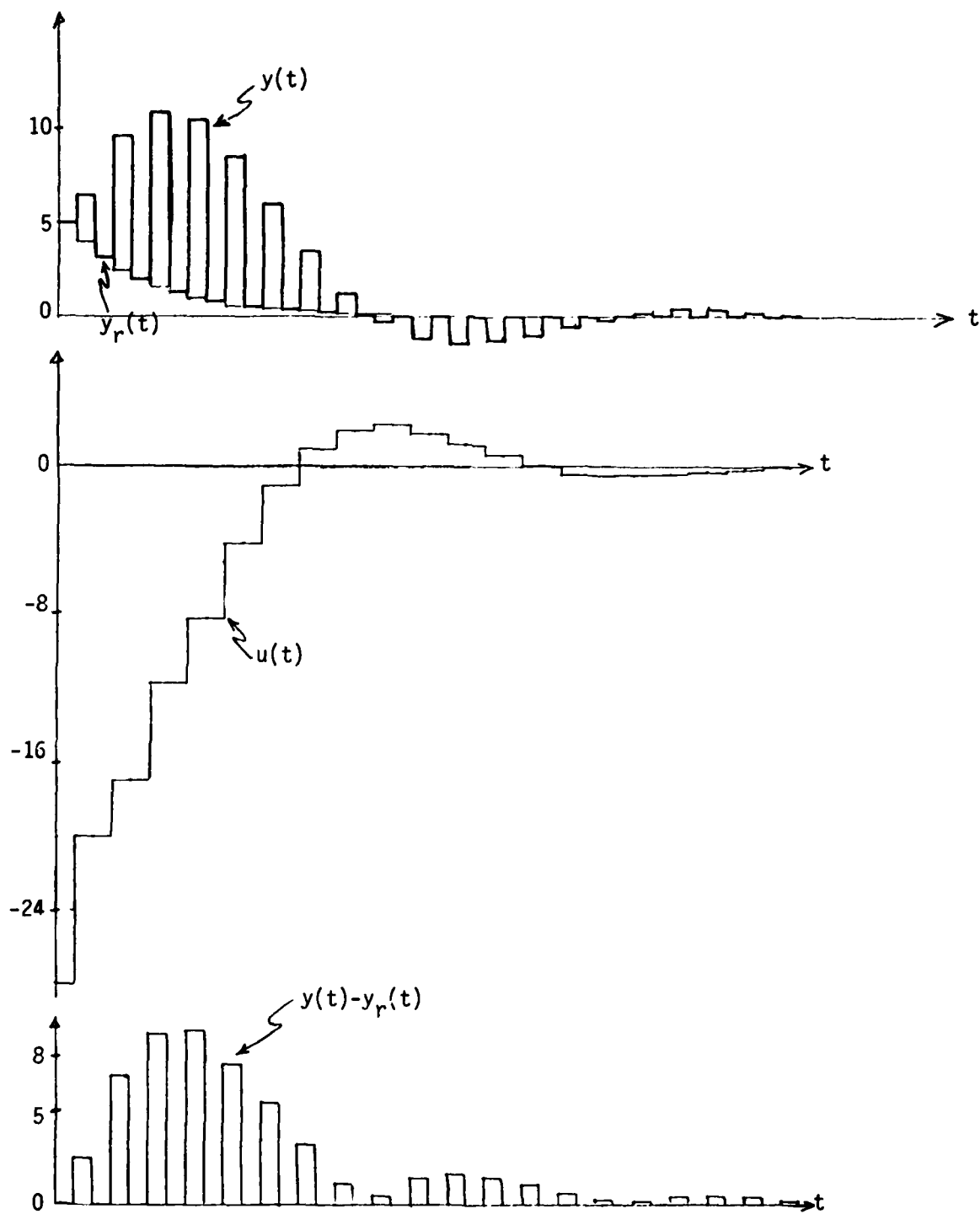
$\underline{h}^T = [1., -2.2, 2.18, -.4420, -1.4303, .6594]. \quad \alpha = .8, \ c = 0.$

Figure 20. Two-step ahead prediction for the nonminimum phase.

One can formulate the p-steps ahead prediction problem with $w_i = 0$ for $i < p$ and $w_p = 1$ as a one-step ahead prediction problem where the N-dimensional previous impulse response $\underline{h}^T$ is represented by a new (N/p)-dimensional impulse response $\underline{g}^T$.

4.6.2   Weighted Distance

In the sequel we will attempt to see how the choice of the weighting factor w_i of equation (4.21) can be used to stabilize a nonminimum phase system. Again, for the sake of simplicity we consider a horizon of prediction of length $p = 2$. We seek an input $u(t)$, constant over two periods, which minimizes the distance

$$J_2 = w_1[y_p(t+1) - y_r(t+1)]^2 + w_2[y_p(t+2) - y_r(t+2)]^2 \qquad (4.31)$$

where y, y_r and y_p are subjected to equations (4.22) through (4.24). We can normalize, without loss of generality, w_1 and w_2 by requiring that

$$w_1 + w_2 = 1$$

The input $u(t)$ which minimizes J_2 is the solution of the following equation, where $u(t+1) = u(t)$:

$$\frac{\partial J_2}{\partial u(t)} = 0$$

which leads to:

$$0 = u(t+1)[w_1 \tilde{h}_1{}^2 + w_2(\tilde{h}_1 + \tilde{h}_2)^2] + \underline{\tilde{h}}^T[w_1 \tilde{h}_1(U-I) + (\tilde{h}_1 + \tilde{h}_2)w_2(U^2-I)]\underline{u}(t-1)$$

$$+ y(t)[\tilde{h}_1(1-\alpha)w_1 + (\tilde{h}_1 + \tilde{h}_2)(1-\alpha^2)w_2] \qquad (4.32)$$

or

$$0 = u(t+1)[w_1 \tilde{h}_1{}^2 + w_2(\tilde{h}_1 + \tilde{h}_2)^2] + w_1 \tilde{h}_1[\underline{\tilde{h}}^T(U-I) + (1-\alpha)\underline{h}^T]\underline{u}(t-1)$$

$$+ w_2(\tilde{h}_1 + \tilde{h}_2)[\underline{\tilde{h}}^T(U^2-I) + (1-\alpha^2)\underline{h}^T]\underline{u}(t-1) \qquad (4.33)$$

The above equation determines the input $u(t)$ which minimizes J_2, as a function of the past input vectors $\underline{u}(t+1)$. It is easy to verify that for $w_1 = 0$, equation (4.33) becomes identical to equation (4.27) which corresponds to two-step ahead prediction. Conversely, for $w_2 = 0$ equation (4.33) is identical with the equation corresponding to the one-step ahead prediction (see Section 4.3).

As in the previous sections, the controller can always determine $(N+1)$ parameters $(w_1, \tilde{h}_1, \ldots, \tilde{h}_N)$ such that the characteristic equation corresponding to equation (4.33) has $N-1$ predetermined poles. However, this solution presents no advantage over the ones described in Section 4.4, and moreover it does not specify the role of w_1.

Let us require that the nathematical model represent the plant exactly; that is, $\underline{h}^T = \underline{\tilde{h}}^T$. Then there remains only one parameter w_1 to adjust for the stabilization of the system. By equating $\underline{\tilde{h}}^T$ to $\underline{h}^T$ in equation (4.33), we deduce:

$$0 = u(t+1)[w_1 h_1{}^2 + w_2(h_1 + h_2)^2] + \underline{h}^T\{w_1 h_1[U - \alpha I] + w_2(h_1 + h_2)[U^2 - \alpha^2 I]\}\underline{u}(t-1)$$

$$(4.34)$$

Clearly, it is not possible to bring all the unstable poles of the above dynamic to predetermined values inside the unit circle by adjustment of only one parameter w_1. However, it might be possible to stabilize the system with a proper choice of w_1; that is, to bring the unstable poles inside the unit circle. It is clear that even this latter attempt is not always assured of success, particularly when the number of unstable poles of $H(z)$ increases. However, experience has shown that in those cases, by increasing sufficiently the length p of the horizon of prediction, it becomes possible to stabilize the system. An intuitive explanation of the above phenomenon is that by increasing the length p, one introduces p-1 additional parameters, w_i, which then can be adjusted to stabilize the system. Comparing the above observation with the approach of Section 4.4, where the horizon of prediction is equal to 1, it appears that the choice of $\tilde{\underline{h}}^T$ or w_i are complementary in the sense that either one fixes the horizon of prediction at $p = 1$ and thus has to determine the appropriate vector $\tilde{\underline{h}}^T$, or one fixes $\tilde{\underline{h}}^T$ ($\tilde{\underline{h}}^T = \underline{h}^T$) and thus has to determine the appropriate length p and the weights w_i ($i = 1,\ldots,p$).

The practical implementation of the above scheme needs minor alteration in the existing MAC software. But the proper choice of w_i needs much acquaintance and experience with the particular system to be stabilized. Figure 21 displays an example of such stabilization for a system of order 5.

$\underline{h}^T = [1., -2.2, 2.18, -.4420, -1.4303, .6594]. \quad \alpha = .6, \ c = 0.$
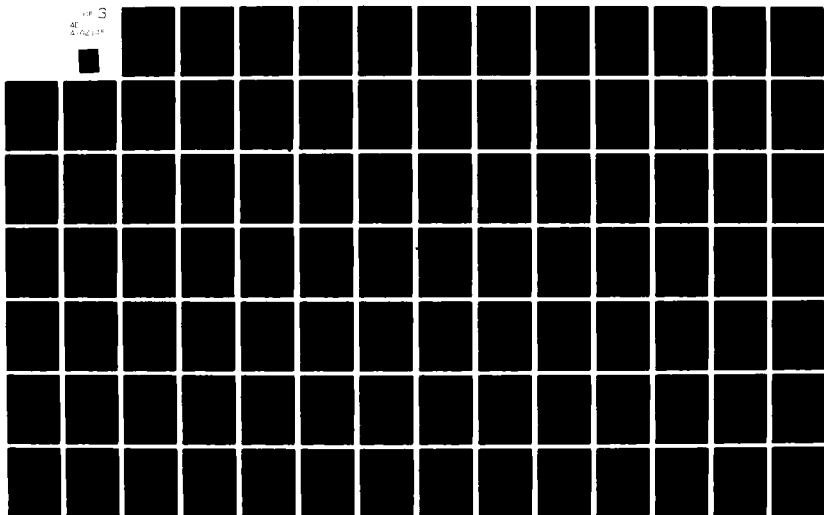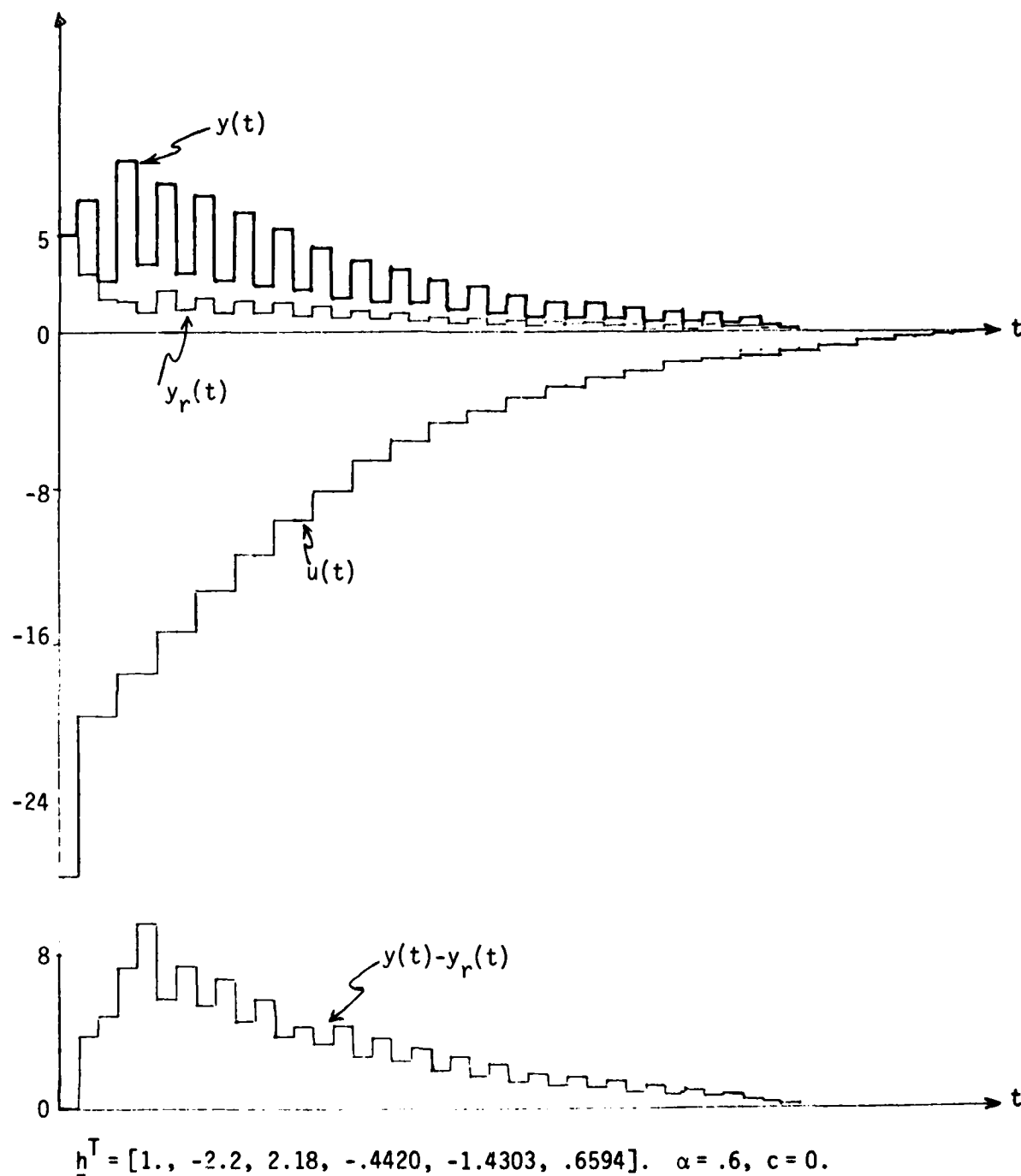
Figure 21.   Two-step ahead prediction with weights $\omega_1 = .2$, $\omega_2 = .8$ for the nonminimum phase system.

4.7 Summary

The problem of nonminimum phase systems has been posed within the framework of MAC. The theoretical analysis has shown that a rather wide range of solutions for the control of such systems exists. These different types of solutions are closely related to each other. Moreover, their implementation needs minor alterations of the regular MAC software.

In practice, the specific solution to choose depends on the particular system and also on the desired response behavior, and generally there is no one "once-through" design method yielding an implementable solution. In most cases, the proper solution is the outcome of a more or less lengthy design process involving both theoretical considerations and also simulation results.

SECTION V

MATHEMATICAL FORMULATION AND ANALYSIS OF
MAC OF CONTINUOUS TIME SYSTEMS

Overview

In this section we develop a mathematical formulation for the MAC
scheme applied to continuous time systems rather than to discrete time
systems as discussed in Section III. This framework provides a pre-
cise formulation of the main components of the control scheme and clearly
indicates the many various possibilities for implementing the basic con-
trol strategy. In addition, this framework provides a means of analyzing
the behavior of the MAC scheme in different situations, and permits an
assessment of its computational requirements. The section is divided
into three sections: the first section presents the general formulation;
the second discusses the optimal solution and the form of the MAC control
law; and the third section briefly discusses the computational aspects
of implementing the MAC control strategy.

5.1 General Formulation of MAC Strategy

Simply stated, the MAC philosophy of control is to choose fictitious
future controls in order to minimize a weighted quadratic distance be-
tween a predicted output, which is a function of the fictitious future
controls, and a desired future output (reference trajectory). The first
step or first few steps of the fictitious future control are actually

applied, and after a short interval of time a new fictitious future control is computed and the process repeats itself. To formulate this precisely, let us make the following definitions.

Predicted future output: $y_p(s,t)$ is the output predicted at future time $t+s$ when t is the present time. Note that $s \geq 0$. The predicted future output $y_p(s,t)$ depends on actual outputs and inputs up to the present time t and on the fictitious future inputs up to the future time $s+t$.

Fictitious future inputs: $u_f(s,t)$ denotes the fictitious future input supposed to be applied at the future time $t+s$ when t is the present time.

Desired future output (reference trajectory): $y_r(s,t)$ denotes the output desired at the future time $t+s$ when t is the present time. This reference trajectory depends on actual outputs and inputs up to the present time t. It may also depend on a prespecified objective function or desired set-point.

The MAC strategy is to minimize a weighted quadratic distance between $y_p(\cdot,t)$ and $y_r(\cdot,t)$ of the form

$$J(u_f(\cdot,t)) = \int_0^\infty |y_p(s,t) - y_r(s,t)|^2 w(s) \ ds \qquad (5.1)^6$$

where J depends on $u_f(\cdot,t)$ through $y_p(\cdot,t)$. In minimizing J we restrict $u_f(\cdot,t)$ to a fixed set U_f of admissible future inputs. The actual input $u(t)$ applied at time t is simply

---

[6] In this discussion the input and output values are finite dimensional vectors. The norm $|v|$ is the usual Euclidean norm for the vector, i.e., $|v|^2 = v_1^2 + \ldots + v_d^2$ where v is d-dimensional and v_i are the coordinates of v with respect to fixed coordinate basis.

84

$$u(t) = u_f*(0,t) \tag{5.2}$$

where $u_f*(\cdot,t) \in U_f$ minimizes J.

To illustrate the control strategy more clearly we consider typical implementations. First, we need to assume a particular functional relationship between the fictitious future control and the predicted future output. In other words, we need a model for the future response of the plant. If we choose a linear model (as we usually do), then y_p can be expressed conveniently as the sum of the zero input response y_{pzi} and the zero state response y_{pzs} of the system

$$y_p(s,t) = y_{pzi}(s,t) + y_{pzs}(s,t) \tag{5.3}$$

That is, $y_{pzi}(s,t) = y_p(s,t)$ if the future input is zero ($u_f(s,t) = 0$ for all $s \geqslant 0$). The zero state response y_{pzs} then gives the future output due to future input alone.

We assume that y_{pzs} is given by an impulse response model in terms of u_f as follows:

$$y_{pzs}(s,t) = \int_0^S H(s-\sigma,t)\, u_f(\sigma,t)\, d\sigma \tag{5.4}$$

In the following discussion we will also assume that $H(s-\sigma,t) = H(s-\sigma)$ is independent of the current time t.

There are several possibilities for representing the zero input response y_{pzi}. For example, one can use the impulse response of the actual past inputs:

85

$$y_{pzi}(s,t) = \int_{-\infty}^{t} H(s-\sigma)u(\sigma) \, d\sigma \qquad (5.5)$$

Alternatively, one might use a state model of the form:

$$y_{pzi}(s,t) = F(s) \, \hat{x}(t) \qquad (5.6)$$

where $\hat{x}(t)$ is a function of actual past inputs and outputs computed by some finite dimensional filter. Other possibilities (e.g., ARMA type models) exist, and one can also consider combinations of these. The important thing to note is that the computations of y_{pzi} and y_{pzs} can be carried out independently. On the one hand, it seems most computationally efficient to use the impulse response model (5.4) to compute y_{pzs}. On the other hand, the impulse response model may not be the most efficient procedure for computing y_{pzi}, and it is a good idea and easy to tailor the zero input response model to the particular problem.

There are also several choices for the reference trajectory y_r. Unfortunately (for purposes of mathematical analysis), these choices are made more on the basis of intuition than systematic theory. However, it is easy to express the most common choices in mathematical terms, and we can subsequently analyze some of the implications of these choices. The most common choice is the first-order reference trajectory of the form

$$y_r(s,t) = e^{-s/\Delta}y(t) + (1 - e^{-s/\Delta})c(t) \qquad (5.7)$$

where Δ is a specified time of response, $y(t)$ is the actual input at time t and $c(t)$ is a set-point or objective at time t (and $c(t)$ is often

constant in this case). Another choice of the reference trajectory is given by

$$y_r(s,t) = c(t+s) \tag{5.8}$$

where $c(\cdot)$ is a pre-specified objective function. (Here $c(\tau)$ usually tends to a constant value as $\tau \to \infty$.) This choice corresponds to the output predictive schemes of Reid et al. (1979, 1980).

Let $y_0(s,t)$ denote the difference

$$y_0(s,t) = y_r(s,t) - y_{pzi}(s,t) \tag{5.9}$$

At time t, we must minimize

$$J(u_f(\cdot,t)) = \int_0^\infty |y_0(s,t) - \int_0^S H(s-\sigma)u_f(\sigma,t)d\sigma|^2 w(s)\,ds \tag{5.10}$$

with respect to $u_f(\cdot,t) \in U_f$. The set U_f of future admissible controls must be a subset of a finite dimensional vector space of inputs (since we have to do the optimization numerically on-line) and it must be such that J has a unique minimum over U_f. Here we assume that U_f can be expressed as a set of unique linear combinations of input functions $u^i(s)$ defined for $s \geq 0$,

$$u_f(\cdot,t) = \sum_{i=1}^n v_i\, u^i(\cdot) \tag{5.11}$$

where $v = (v_1, v_2, \ldots, v_n)^T \in V$ and V is a closed, bounded polyhedron in $\mathbf{R}^n$.[7] The optimization of J is performed in $\mathbf{R}^n$ on the finite dimensiorl

---

[7]Here and elsewhere $(\ )^T$ denotes the transpose of both vectors and matrices.

vectors V. In terms of $v \in V$, J is a quadratic form in v,

$$J(v) = \sum_{j=1}^{n} \sum_{k=1}^{n} P_{jk} v_j v_k - 2 \sum_{j=1}^{n} q_j(t) v_j \qquad (5.12)$$

where

$$q_j(t) = \int_0^\infty w(s) y_0(s,t)^T \left\{ \int_0^S H(s-\sigma) u^j(\sigma) d\sigma \right\} ds \qquad (5.13)$$

and

$$P_{jk} = \int_0^\infty w(s) \left\{ \int_0^S \int_0^S u^j(\sigma)^T H(s-\sigma)^T H(s-\tau) u^k(\tau) d\sigma d\tau \right\} ds \qquad (5.14)$$

where $j,k = 1,2,\ldots,n$. Note that P_{jk} is independent of current time t and doesn't need to be computed on-line. In order to ensure a unique minimum, we assume $u^i(\cdot)$ are chosen so that $[P_{jk}]$ is a nonsingular matrix (hence positive definite).

The linear terms $q_j(t)$ do depend on t through $y_0(s,t)$--that is, due to the predicted zero input response $y_{zpi}(\cdot,t)$ and the reference trajectory $y_r(\cdot,t)$. However, we can often arrange the computation of $q_j(t)$ so that most of the calculations are done off-line. For example, if y_{pzi} is given by (5.6) and y_r is given by (5.7), then $q_j(t)$ can be written

$$q_j(t) = \hat{x}(t)^T q_j^1 + y(t)^T q_j^2 + c(t)^T q_j^3 \qquad (5.15)$$

where q_j^1, q_j^2, q_j^3 are vectors which can be computed off-line as

$$q_j{}^1 = -\int_0^\infty w(s)F(s)^T\left\{\int_0^s H(s-\sigma)u^j(\sigma)d\sigma\right\}ds \qquad (5.16)$$

$$q_j{}^2 = \int_0^\infty w(s)\ e^{-s/\Delta}\left\{\int_0^s H(s-\sigma)u^j(\sigma)d\sigma\right\}ds \qquad (5.17)$$

$$q_j{}^3 = \int_0^\infty w(s)\ (1 - e^{-s/\Delta})\left\{\int_0^s H(s-\sigma)u^j(\sigma)d\sigma\right\}ds \qquad (5.18)$$

At this point let us review the main steps of the control algorithm:

MAC Strategy

(i)   Compute certain terms, such as P_{jk} in (5.14) and q_j^ℓ in (5.16) - (5.18) off-line.

(ii)   At each control update time t, compute $q_j(t)$ from (5.13) or (5.15).

(iii) Minimize the quadratic form $J(v)$ in (5.12) with respect to v and its linear inequality constraints. Let $v^*(t)$ denote the minimum solution.

(iv)   Apply the actual control input

$$u(t) = \sum_{k=1}^n v_k{}^*(t)u^k(0) \qquad (5.19)$$

up until the next control update time.[8]

(v)   Return to (ii) and repeat.

5.2 Solution of Optimization and Form of MAC Control Law

The type of control law generated by the MAC strategy depends on the way in which the terms $v_k{}^*(t)$ in (5.19) depend on the past data. In turn,

---

[8]More generally, one would have $u(t+s) = \sum_{k=1}^n v_k^*(t)u^k(s)$ for times t+s up until the next update.

89

the dependence of $v_k*(t)$ on the past data is determined by the dependence of $q_j(t)$ in (5.12) on the past data. Note that $q_j(t)$ depends linearly on the past in the cases we considered in the previous section. If the minimum $\{v_k*(t)\}$ is unconstrained, then it too depends linearly on the past. Let $v*(t)$ denote the n-vector with coordinates $q_k(t)$ and let P denote the $n \times n$ matrix $[P_{kj}]$. Then, the unconstrained minimum of $J(v)$ in (5.12) is given by

$$v*(t) = P^{-1}q(t) \tag{5.20}$$

The control law (5.19) can be written

$$u(t) = Uv*(t) \tag{5.21}$$

where $u^k(0)$ is the $k\underline{th}$ column of the matrix U. In case of unconstrained $v*(t)$ we get the linear control law

$$u(t) = UP^{-1}q(t) \tag{5.22}$$

If the reference trajectory is given by (5.7) and the zero input response is given by (5.6), then (5.15) is true and (5.22) can be rewritten explicitly in terms of the past data as

$$u(t) = UP^{-1}(Q^1\hat{x}(t) + Q^2y(t) + Q^3c(t)) \tag{5.23}$$

In this expression Q^ℓ is the matrix with $(q_j^\ell)^T$ as its $j\underline{th}$ row.

In case $v*(t)$ is constrained, the control law is nonlinear and *much* more complicated. However, something more can be said about the

<u>functional form</u> of the control law. Consider the figure below (Figure 22). In this case $n = 2$ and the constraints on v_1 and v_2 have the form

$$v_k^- \leqslant v_k \leqslant v_k^+, \qquad k = 1,2 \qquad (5.24)$$

The constraint set V is the square region indicated in the figure. The dashed lines in the figure represent directions orthogonal to the constraints $v_k = v_k^{\pm}$. The orthogonality of the directions is with respect to P and in general these directions would be oblique to the constraints. For simplicity we have drawn them perpendicular (this corresponds to a diagonal P).

Let $v^{**}(t)$ denote the unconstrained minimum

$$v^{**}(t) = P^{-1}q(t) \qquad (5.25)$$

which may give $v^{**}(t) \notin V$. It is not hard to see that in each of the regions 0, 1, 2, 3, 4, 5, 6, 7, 8 shown, the constrained minimum $v^*(t)$ is an affine function of the unconstrained minimum $v^{**}(t)$. That is to say, $v^*(t)$ can be written as

$$v^*(t) = \phi'(v^{**}(t))v^{**}(t) + \phi'(v^{**}(t)) \qquad (5.26)$$

where $\phi'(v^{**})$ is an $n \times n$ matrix which is piecewise constant on the regions 0 - 8, and $\phi'(v^{**})$ is a piecewise constant n-vector on the same regions. Thus, we say that the general MAC control law given by (5.21), (5.25) and (5.26) is piecewise linear-affine. The pieces are especially nice, being defined by simple linear inequalities. Unfortunately, as n grows, the number of pieces grows exponentially. Nevertheless, there may be effective
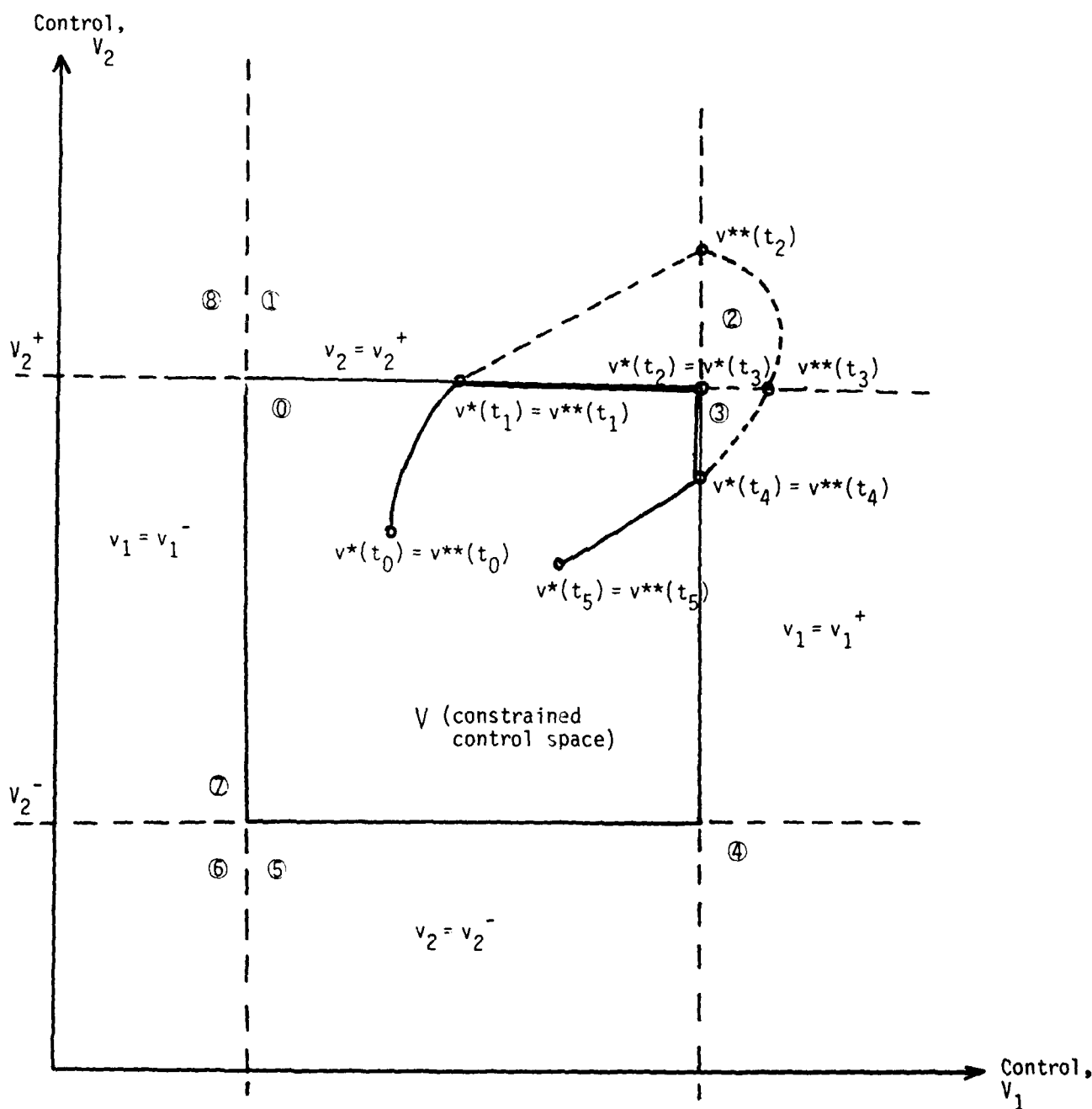
91

Figure 22.   Illustration of nonlinear (piecewise linear affine) control law.

methods of analyzing this type of control law. For instance, it is an example of the <u>variable structure systems</u> studied by Utkin (1977). Also in certain cases it might be possible to analyze the behavior of $v^*(t)$ effectively in terms of the unconstrained minimum $v^{**}(t)$. However, it must be remembered that in general $v^{**}(t)$ will evolve according to a nonlinear law also. When the inequality constraints are not as simple as in (5.24) (for example, if there are state constraints), the situation becomes much more complex than pictured in Figure 22. It is still possible to express the control law as piecewise linear-affine, but the pieces become much more complicated. In any case, we will not pursue the investigation of the nonlinear control law any further here. Appendix A investigates the nonlinear type control law further for single input-single output systems.

So far we have seen that the general MAC control law is nonlinear and quite difficult to analyze.[9] The MAC control law is linear in the case that there are no inequality constraints on the set of admissible future controls U_f so that $V = \mathbf{R}^n$. However, the linear case offers its own difficulties due to the possibility of unbounded control inputs (since J does not weight controls at all).[10] The choice of U_f and the reference trajectory in this case can make the difference between stable and unstable, robust and non-robust control.

Before finishing this section, we note that the most common choice of U_f is the set of inputs piecewise constant over disjoint intervals of time. Thus, the basis inputs $u^i(\cdot)$ have the form

---

[9] although the particular piecewise linear-affine structure offers some promise of analysis.

[10] Control weights are easily added and easily analyzed, of course, but their use has not been required in the examples we have treated.

$$u^i(s) = u^i, \qquad s \in I_i$$
$$= 0, \qquad s \notin I_i \tag{5.27}$$

where u^i is a constant vector and I_i is a bounded interval in $[0,\infty)$. The choice of I_i (the so-called "time blocks" or "blocking") greatly affects the type of unconstrained control computed by the MAC control law. Such a choice of admissible future controls U_f leads one naturally to analyze the problem in terms of discrete time models as presented in Section III.

5.3 Computational Considerations

In case of input or output constraints, one must use a numerical optimization algorithm to minimize the quadratic form (5.12) with respect to the constraints. Currently, a simple gradient projection algorithm (Canon, Callum and Polak, 1970) is used to solve this optimization problem. The problem is set up as a quadratic optimization with linear inequality constraints on the vector v appearing in (5.12). Note that if there are constraints specified on the input $u(\cdot)$, then these usually show up as fairly simple constraints on v. Output[11] constraints, on the other hand, show up as more complicated linear inequality constraints on v, and these tend to slow down the computation of the optimal v*. If only input constraints are present, the complexity of the optimization problem depends only on the dimension n of the matrix P in (5.12). In most cases this dimension is small ($n \leqslant 20$). However, output constraints can seriously increase the complexity of the problem even if P has small dimension. In this case it would be advisable to use a small number of constraints on v

---

[11]State constraints can be formulated as (unobserved) output constraints in this framework.

94

to approximate the desired output constraints.

Another important aspect of *computation* is how much must be computed on-line and updated each time one wants a new control. As noted above, certain quantities such as P_{jk} and q_j^ℓ in (5.14), (5.16) - (5.18) can be computed off-line and stored for on-line use. Input constraints appear as simple linear inequalities on the v_j which are usually independent of time. Output constraints, on the other hand, may be time-dependent when expressed in terms of linear inequalities in the variables v_j.

Although there are a number of sophisticated algorithms for solving quadratic optimization problems with linear inequality constraints, we are more concerned at present with obtaining a clear understanding of the behavior of MAC controlled systems. For this reason, we have restricted the current research to a simple gradient projection algorithm. Nevertheless, it is good to note that this simple algorithm is sufficiently fast to update the control inputs for the problems considered so far.

SECTION VI

MAC FOR CONTINUOUS-TIME SYSTEMS WITH
DISCRETE OBSERVATIONS AND UNOBSERVED OUTPUTS

Overview

In this section, we investigate MAC for an incompletely observed continuous linear time invariant system containing both process and measurement noise. The performance index is taken to be the expected value of a quadratic function of the continuous tracking error and control effort. In accordance with LQG theory, the subject is analyzed as two essentially separable problems: the estimation and the control. The estimation, which is of minimum variance type, is done by Kalman filtering. For the control part, two types of solutions are considered. One is based on the usual MAC approach utilizing impulse response functions and the other is based on state feedback control. The latter approach is taken here to study the sample rate problem, i.e., determination of the effect of sample rate on controller performance. The organization of this section is as follows. The problem is formally stated and discussed in Section 6.1. Section 6.2 analyzes the optimal estimation problem. Given the output of the estimator, the optimal control is discussed in Section 6.3. Section 6.4 is devoted to analyzing the effect of sample rate. The plant model in this section is assumed to be in a state vector form. This is convenient for studying the problem of unobserved outputs and sample rate selection. In addition, for aerospace vehicles, state vector models are generally

available. MAC implementation, however, is not restricted to purely impulse response models or state vector models. As shown in Section V, it is possible to use the state model for estimation-prediction purposes and the impulse response model for control computations. Since these approaches are mathematically equivalent for the idealized case of unconstrained inputs, we will formulate the sample rate selection problem in terms of the state vector approach, thereby exploiting the results of LQG theory.

6.1 Statement of the Problem

Consider a discretely observed continuous-time linear time-invariant system, given by

$$\dot{\underline{x}}(t) = F\underline{x}(t) + G\underline{u}(t) + \underline{w}(t)$$
$$\underline{y}(t) = H\underline{x}(t) + \underline{v}(t)$$

(6.1)

where

$$\underline{x}(\cdot) \in R^n, \quad \underline{y}(\cdot) \in R^m, \quad \underline{u}(\cdot) \in R^r;$$

$$E[\underline{x}(t_0)] \triangleq \bar{\underline{x}}(t_0);$$

$$E[(\underline{x}(t_0) - \bar{\underline{x}}(t_0))(\underline{x}(t_0) - \bar{\underline{x}}(t_0))^T] \triangleq P_x(t_0);$$

$$F \in R^{n \times n}, \quad G \in R^{n \times r}, \quad H \in R^{m \times n};$$

$$\underline{w}(\cdot) \in R^n, \quad \underline{v}(\cdot) \in R^m;$$

$\underline{w}(\cdot)$ and $\underline{v}(\cdot)$ are zero-mean, uncorrelated white noise vectors with:

$$E[\underline{w}(t)\underline{w}^T(t+\tau)] \triangleq Q(t)\delta(\tau);$$

$$E[\underline{v}(t)\underline{v}^T(t+\tau)] \triangleq R(t)\delta(\tau);$$

$$Q(\cdot) \in R^{n \times n}, \quad Q(\cdot) \geqslant 0; \quad R(\cdot) \in R^{m \times m}, \quad R(\cdot) > 0.$$

The system is observed at discrete times t_i, $i = 1, \ldots, p$:

$$\underline{y}(t_i^+) = H\underline{x}(t_i^+) + \underline{v}(t_i^+) \tag{6.2}$$

It is desired to determine the optimum control vector $\underline{u}(\cdot)$ which minimizes the cost

$$J = E[\tfrac{1}{2} \int_{t_0}^{t_0+T} (\|\underline{y}(t) - \underline{y}_r(t)\|^2 + \rho\|\underline{u}(t)\|^2)dt] \tag{6.3}$$

where $\underline{y}_r(\cdot)$ is the reference trajectory. It is assumed that $\underline{y}_r(\cdot)$ has first order dynamics, and that the setpoint is at the origin. Then

$$\dot{\underline{y}}_r(t) = -\frac{1}{\tau} \underline{y}_r(t) \tag{6.4}$$

and $\qquad \underline{y}_r(t_0) = \underline{y}(t_0)$

and the reference trajectory is given as a function of the initial observation and time as

$$\underline{y}_r(t) = \underline{y}(t_0)e^{-((1/\tau)(t-t_0))} \tag{6.5}$$

or

$$\underline{y}_r(t) = \alpha \underline{y}(t_0)$$

where

$$\alpha = e^{-(t-t_0)/\tau}$$

The positive constant ρ in equation (6.3) is the cost associated with the input energy.

It is assumed that at time $t \in [t_0, t_0+T]$, the controller has the information $Y(t) = \{\underline{y}(t_i^+); \ t_0 \leqslant t_i \leqslant t\}$, i.e., all samples prior to t. The determination of $\underline{u}(t)$ is based on the information set $Y(t)$.

It is well known that the determination of the optimal input $\underline{u}(\cdot)$ involves an estimation problem and a control problem. In the remaining part of the present section, the separation property (separation theorem) is derived for the system given by equations (6.1) and (6.2).

The cost J can be rewritten as:

$$J = E\{E[\tfrac{1}{2} \int_{t_0}^{t_0+T} (\|\underline{y}(t) - \underline{y}_r(t)\|^2 + \rho\|\underline{u}(t)\|^2)dt \,|\, Y(t_0+T)]\} \qquad (6.6)$$

where $E[\ldots Y(t)]$ denotes the conditional expectation given an information set $Y(\cdot)$, while $E\{\ldots\}$ denotes the expectation over all possible information sets. Then

$$J = E\{\tfrac{1}{2} \int_{t_0}^{t_0+T} (E[\|\underline{y}(t) - \underline{y}_r(t)\|^2 \,|\, Y(t_0+T)] + \rho E[\|\underline{u}(t)\|^2 \,|\, Y(t_0+T)])dt\} \qquad (6.7)$$

99

The conditional mean and covariance of the random vector $\underline{y}(t)$ are given by:

$$E[\underline{y}(t)|Y(t)] \triangleq \hat{\underline{y}}(t)$$

$$E[(\underline{y}(t) - \hat{\underline{y}}(t))(\underline{y}(t) - \hat{\underline{y}}(t))^T|Y(t)] \triangleq P_y(t)$$

Then

$$E[\|\underline{y}(t) - \underline{y}_r(t)\|^2|Y(t)] = \|\hat{\underline{y}}(t) - \hat{\underline{y}}_r(t)\|^2 + \text{tr } \text{cov}[\underline{y}(t) - \underline{y}_r(t)] \tag{6.8}$$

where $\hat{\underline{y}}_r(\cdot)$ is defined similarly.

The control being independent of the observation noise:

$$E[\|\underline{u}(t)\|^2|Y(t)] = \|\underline{u}(t)\|^2 \tag{6.9}$$

it remains to evaluate $\text{cov}[\underline{y}(t) - \underline{y}_r(t)]$ using

$$\text{cov}[\underline{y}(t) - \underline{y}_r(t)] = \text{cov}[\underline{y}(t)] + \text{cov}[\underline{y}_r(t)] - 2E[(\underline{y}(t) - \hat{\underline{y}}(t))^T(\underline{y}_r(t) - \hat{\underline{y}}_r(t))] \tag{6.10}$$

It follows that:

$$J = E[\tfrac{1}{2}\int_{t_0}^{t_0+T}(\|\hat{\underline{y}}(t) - \hat{\underline{y}}_r(t)\|^2 + \rho\|\underline{u}(t)\|^2)dt + \tfrac{1}{2}\text{tr}\int_0^T \text{cov}[\underline{y}(t) - \underline{y}_r(t)]dt] \tag{6.11}$$

where

$$\text{cov}[\underline{y}_r(t)] = P_{y_r}(t) = E[H^T \underline{x}^T(0)\underline{x}(0)H]e^{-2(1-\alpha)(t-t_0)}$$

$$P_{y_r}(t) = H^T P_x(t_0)He^{-2(1-\alpha)(t-t_0)}$$

$P_{y_r}(t)$ is independent of the estimation and control strategy. The task hence reduces to minimizing

$$J = E[\tfrac{1}{2}\int_{t_0}^{t_0+T}(\|\hat{\underline{y}}(t) - \hat{\underline{y}}_r(t)\|^2 + \rho\|\underline{u}(t)\|^2 + \text{tr } P_y(t) - 2E[(\underline{y}(t) - \hat{y}(t))^T$$

$$\cdot(\underline{y}_r(t) - \hat{\underline{y}}_r(t))|Y(t)])dt] \tag{6.12}$$

where $P_y(t)$ denotes the covariance of $\underline{y}(t)$ conditional on the information $Y(t)$.

The above development shows that two problems must be solved; first, it is necessary to design an estimator to obtain $\hat{\underline{y}}(t)$:

$$\hat{\underline{y}}(t) = E[\underline{y}(t)|Y(t)]$$

This estimator is of the minimum variance type. The second, in a sense independent, problem is to derive the control $\underline{u}$ that minimizes equation (6.12). The following section deals with the design of the estimator.

6.2 Optimal Estimation

The optimal (in the minimum-variance sense) estimate $\hat{\underline{y}}(t)$ of the system (6.1) given the discrete observations $y(t_i^+)$, $i = 1,\ldots,p$, is generated

101

(see Jazwinski, 1970 and Bryson and Ho, 1975) by the linear filter:

$$\dot{\hat{\underline{x}}}(t) = F\hat{\underline{x}}(t) + G\underline{u}(t) + K(t_i)(\underline{y}(t_i^+) - H\hat{\underline{x}}(t_i^-))\delta(t-t_i)$$

$$(6.13)$$

$$\hat{\underline{y}}(t) = H\hat{\underline{x}}(t)$$

for $t \in [t_i, t_{i+1})$, $i = 1, \ldots, p$. Here $K(t_i)$ is the Kalman gain and the term

$$(\underline{y}(t_i^+) - \hat{\underline{y}}(t_i^-)) = (\underline{y}(t_i^+) - H\hat{\underline{x}}(t_i^-)) \qquad (6.14)$$

is the innovation, which contains the new information due to the observations at time t_i, $\underline{y}(t_i^+)$. It may be noted that in general

$$\underline{y}(t_i^+) \neq \hat{\underline{y}}(t_i^-) = \lim_{t \to t_i^-} H\hat{\underline{x}}(t) \qquad (6.15)$$

The system (6.13) describes the evolution of the estimate $\hat{\underline{y}}(t)$ between instantaneous observations at times t_i. The Kalman gain $K(t_i)$ is obtained in terms of the solution of a Riccati equation. Since these results are well documented elsewhere (see, e.g., Astrom, 1970, Kwakernaak and Sivan, 1972), we omit the details here.

6.3 Optimal Control

Once the complete state of the system has been estimated, all the controlled outputs (observed and unobserved) can be predicted for different fictitious inputs in the future (see Section V). One may use either the state vector model or the impulse response model for predicting the zero state response, depending on the dimensionality of each representation and the associated computational effort. The optimization

problem, however, is more efficiently solved using a discretized impulse response representation, particularly when hard constraints have to be satisfied.

The computation of the optimal controls using the impulse response model is similar to that carried out in Section 5.1. The certainty equivalence principle (Astrom, 1970), however, does not apply strictly in the constrained control (or state) case. The analysis of this difficult case requires further investigation. We assume here that the effect of constraints is taken into account indirectly by the control effort weighting term in the performance index. The optimal control problem is thus equivalent to an LQG problem, which may be stated in continuous-time or discrete-time depending on the nature of the control input. Most commonly, the control input is kept constant (or linearly interpolated) between sample points so that a discrete-time formulation is appropriate. Notice that the sample rate for the control input application need not be the same as the sample rate for output observation. The former restricts the class of inputs over which optimization is performed, whereas the latter influences the prediction accuracy. The effect of these two sample rates on controller performance can be studied in terms of the solution to two discrete-time Riccati equations, the control Riccati equation obtained by discretizing the system equations and the performance index at the controller sample rate, and the estimation Riccati equation obtained by discretizing the system equations at the observation sample rate. The relevant equations are given in Kwakernaak and Sivan (1972), pp. 542-550. (See, in particular, equations 6-486, 6-487, 6-488, 6-523, 6-524 and 6-525.)

6.4 Summary

It was shown that the problem of MAC design for stochastic continuous-time systems with noisy incomplete discrete-time observations can be reduced to the solution of an estimation problem and a deterministic control problem. The former is solved in terms of a Kalman filter (see Section VII for an alternative approach) and the latter is shown to be solvable either using LQG theory or the procedure outlined in Section V. A study of the sample rate selection problem is indicated in terms of the solution of two Riccati equations.

104

SECTION VII

RECURSIVE PREDICTORS

Overview

This section describes the preliminary development of an estimator to work with IDCOM in the control of systems in noisy environments. The results of Richalet et al. (1978), Mehra et al. (1978) and our preliminary simulation results indicate that IDCOM works well in low-noise environments. In several interesting processes, however, this ideal environment cannot be guaranteed. A state estimator (Kalman filter) can of course be included in IDCOM, for output prediction, when a good state model is available. But the requirement of a good state model is a severe one, and is not shared by the control optimization part of IDCOM. We were thus led to consider alternate predictors which are compatible with IDCOM yet offer good performance when noise and model uncertainty are present.

This section begins with a description of the current predictor in IDCOM (Section 7.1). Section 7.2 then discusses a simple, useful modification which improves performance, and Section 7.3 compares this predictor with the usual one. Autoregressive Moving Average (ARMA) models are briefly discussed in Section 7.4, and optimal estimation is related to this formulation in Section 7.5.

7.1 IDCOM Prediction

It is useful to note how IDCOM *currently computes* the state (output)
predictions that it requires for control computation. The IDCOM fun -
tions may be outlined as in Figure 23. The current output, $y_a(t)$, is fed
to two IDCOM blocks, one of which calculates the reference trajectory
for several future time steps $y_r(s,t)$[12] and one which calculates the pre-
dicted output for zero-input in the future (i.e., no future controls
applied) $y_{pzi}(s,t)$.
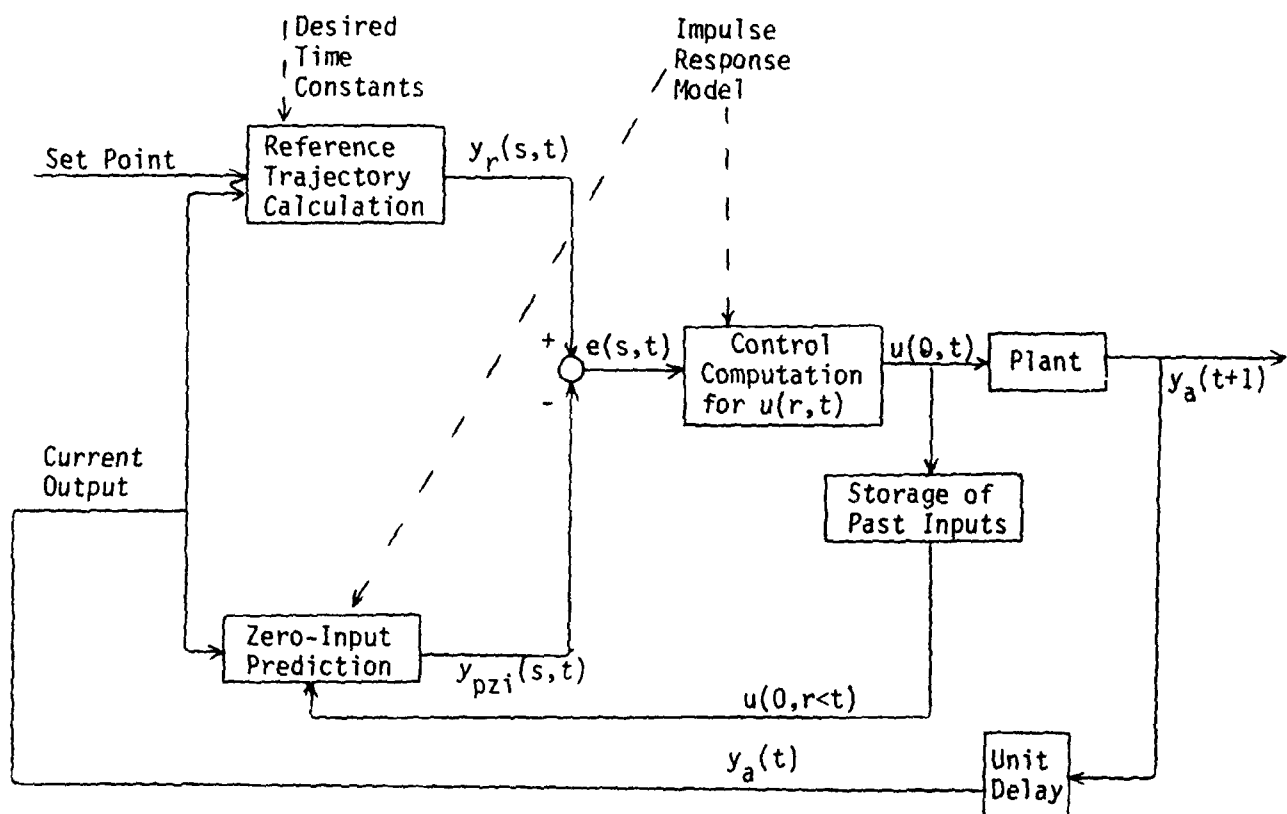
Figure 23. IDCOM functional diagram.

---

[12] *This notation is the same as that of Section V.*

The difference between these terms is the sequence $e(s,t)$ of future errors to be removed by the future controls, $u(r,t)$. The only control actually applied is $u(0,t)$, the current choice, since the calculation is repeated after the next step.

The internal (to IDCOM) impulse response model is usually used in two places--the control calculation and the zero-input prediction. These two functions do not require the same model, and in many cases an alternate prediction model (e.g., for noisy environments) or control model (e.g., for nonminimum phase systems) may be desirable.

In general, however, IDCOM uses the same impulse response model, identified (off- or on-line) for both functions. In order to avoid the dangers of "open loop" prediction, however, where only the past inputs and internal model are used, IDCOM modifies the standard prediction by adding a bias-compensation term to improve performance. Thus, where the "open loop" prediction for n-steps ahead would be[13]

$$y_{pzi_{OL}}(n+1,t) = \sum_{i=0}^{T-n-1} H_{i+n}u(t-i)$$

the "closed loop" (bias compensated) version is

$$y_{pzi_{CL}}(n+1,t) = \sum_{i=0}^{T-n-1} H_{i+n}u(t-i) + y_a(t) - y_{pzi}(0,t) \qquad (7.1)$$

where

$$y_{pzi}(0,t) = \sum_{i=1}^{T} H_i u(t-i) \qquad (7.2)$$

---

[13]This assumes that only T impulse response terms are stored.

is the estimated current output based on the past inputs and the internal model. Intuitively, the "closed loop" predictor applies the bias between the actual and currently predicted outputs to all future outputs. This technique has worked quite well for standard, stable plants (with finite impulse response).

7.2 Modification for Unstable (Integrator) Plants

While investigating the application of IDCOM to simple systems (e.g., integrators) with infinite impulse responses, it was discovered that a slight modification to the above scheme resulted in a predictor which could tolerate impulse responses which merely became constant (rather than zero) for large T. By recomputing $y_{pzi}(0,t)$ at each time $(n+1)$ for which a prediction is needed, an expression for $y_{pzi_{CL}}$ can be obtained which converges whenever

$$H_{i-n} - H_i = 0 \qquad \forall i > T \qquad (7.3)$$

rather than the usual requirement that

$$H_i = 0 \qquad \forall i > T \qquad (7.4)$$

Specifically, we choose

$$y_{pzi_{new}}(n+1,t) = \sum_{i=0}^{T-n-1} (H_{i+n} - H_i)u(t-i) + y_a(t)$$

which results from changing the upper limit in equation (7.2).

7.3 Comparison of Predictors

We can compare the behavior of these two predictors by considering the actual output $y_{zi}(u,t)$ resulting from zero inputs after time t. We define the prediction error for the new predictor as

$$e(n,t) = y_{pzi}(n,t) - y_{zi}(n,t)$$

Then

$$e(n+1,t) = \sum_{i=0}^{T-n-1} (H_{i+n} - H_i)u(t-i) - [y_{zi}(n+1,t) - y_a(t)]$$

We note that if H from the internal model is exactly the real H, then $e(s,t) = 0$ for all s as long as H satisfies equation (7.3). Thus, even if H does not become zero for large T, good prediction is possible. By comparison, the error in the regular computation would be

$$e(n+1,t) = \sum_{i=0}^{T-n-1} (H_{i+n} - H_i)u(t-i) - [y_{zi}(n+1,t) - y_a(t)] - \sum_{T-k-1}^{T-1} H_i u(t-i)$$

$$= - \sum_{T-n-1}^{T-1} H_i u(t-i)$$

which, in general, is not zero.

Similar expressions can be obtained for cases of bias and scale factor errors in the impulse response. If the internal model (H') is equal to the real impulse response plus a bias, i.e.,

$$H_i' = H_i + b$$

109

then the above error becomes 0 for the new technique and

$$e(n+1,t) = \sum_{T-n-1}^{T-1} (H_i + b)u(t-i)$$

for the regular predictor. For scale factor mismatch,

$$H_i' = qH_i$$

then

$$e(n+1,t) = (q-1) \sum_{i=0}^{T-n-1} (H_{i+n} - H_i)u(t-i)$$

$$= (q-1)(y_{zi}(n,t) - y_a(t))$$

for the new technique, which results in an error which grows with prediction distance (i.e., as n grows). The regular predictor has this error plus another term:

$$e(n+1,t) = (q-1)(y_{zi}(n,t) - y_a(t)) - \sum_{T-n-1}^{T-1} qH_i u(t-i)$$

In general, it appears that the modified technique is better under most circumstances, with only a slightly greater computation cost (re-computing many "current estimates" based on prediction lag). This method is now used on most of our programs. We note that the predictor for n steps ahead is always written as a function of the current observation, and not as a function of the n-1st prediction. This preserves the separation between prediction and control computations, since the future inputs would be required for the full estimate of the n-1st term.

110

7.4 ARMA Models

The use of past inputs and impulse responses for output prediction corresponds to a "moving average" filter (see Box and Jenkins (1976) for a full discussion of this topic). The inclusion of the current observation creates a mixed, autoregressive-moving average (ARMA) predictor as often used in time series analysis. It was noted in the last section that the inclusion of the current observation permitted the treatment of infinite-impulse response models. Indeed, this is a familiar fact in ARMA modeling, and often motivates the use of mixed models.

In our current version of IDCOM, a 50th order impulse response (moving average) is used with a first order autoregressive term. It seems probable that a lower order mixed-model would give equally good (or better) prediction with less chance of numerical problems, straightforward identification, and the ability to handle measurement noise. Such low-order ARMA models combined with a short impulse response control calculation would, we believe, provide good control and estimation for systems in noisy environments while preserving the robustness properties of MAC. We intend to investigate this area in the near future.

7.5 Optimal Estimation

Before ending this section, we wish to briefly outline the solution of an optimal estimation problem for IDCOM, with connections to ARMA models and simplified estimation. For convenience, we begin with a continuous-time plant with white Gaussian process (w) and measurement (v) noise, and dynamics:

$$\dot{x}(t) = Ax + Bu + w(t)$$

$$y = Cx$$

$$z(t) = y(t) + v(t)$$

where x is the state, y the output, and u a known input. The standard
Kalman filter for this system would be

$$\dot{\hat{x}} = A\hat{x} + Bu + K(z - C\hat{x})$$

$$= (A - KC)\hat{x} + Bu + Kz$$

where K is the gain matrix. In steady state, and with $\hat{x}(0) = 0$, this system
can be represented as

$$\hat{x} = \int_0^t \phi_{CL}(t,\tau)[Bu + Kz]\, d\tau$$

and

$$\hat{y} = \int_0^t C\phi_{CL}(t,\tau)[Bu + Kz]\, d\tau$$

where ϕ is the closed-loop transition matrix

$$\phi_{CL}(t,\tau) = e^{(A-KC)(t-\tau)}$$

The estimate y of the output can be approximated in discrete-time by

$$\hat{y}(N\Delta) = \sum_0^N C\phi_{CL}[Bu + Kz]\Delta$$

112

which can be written

$$\hat{y}(N\Delta) = \sum_{i=0}^{N} H_1(i)u_{N-i} + \sum_{i=0}^{N} H_2(i)z_{N-i}$$

where $H_1 = C\phi_{CL}B\Delta$ and $H_2 = C\phi_{CL}K\Delta$. Without noise, $K = 0$, and y is given by

$$y(t) = \int_0^t C\phi_{OL}(t,\tau)Bu(\tau)d\tau + y(0)$$

where

$$\phi_{OL}(t,\tau) = e^{A(t-\tau)}$$

and

$$\hat{y}(N\Delta) = \sum_{i=0}^{N} H_{OL}(i)u_{N-i}$$

where $H_{OL} = C\phi_{OL}B\Delta$. Thus, we note the similarity between noise-free, "open loop" prediction based on u alone and closed-loop prediction based on both inputs and measurements. In general, there is no "easy" relationship between H_{OL} and H_1 and H_1, although straightforward calculations yield the latter two.

In applying these results to IDCOM, one is forced to create 50th order state systems to handle an arbitrary 50th order impulse response. The calculations for such large systems are awkward at best, and the identification of lower-order ARMA models would greatly simplify the filter computations.

113

Finally, we suspect that the combination of exponentially-aged filters with reduced-order ARMA plant models may prove to be quite workable and practical. Exponential aging (see Miller, 1971, or Schweppe, 1973), sometimes called exponential least squares, is a method of filter design which exponentially weights the past data. By introducing an exponential aging term on the measurement noise covariance, the Kalman filter gains remain open (non-zero) and the filter accepts new information even without a process noise term.

Currently, the construction of noise covariances is often fictitious, done merely to keep the filter gains in a region which results in good performance. An alternate approach, somewhat less arbitrary, would require control system designers to specify only measurement noise covariances (measured or from sensor specifications) and an exponential weighting factor (based on desired bandwidth but explainable as easily as the reference trajectory time constant of IDCOM). Such specifications, along with a low order ARMA process model, may result in a simple estimator with adequate performance and the ability to be easily modified by test personnel as sensors change or experience with the estimator increases.

7.6 Summary

This section has described our preliminary work in improving output prediction for IDCOM. An improved predictor was described for normal operation with model uncertainty. Techniques for designing predictors in noisy environments were discussed, and directions for future research were given.

SECTION VIII

MAC APPLICATION TO A MISSILE CONTROL PROBLEM

Overview

This section describes the application of MAC to a missile attitude control simulation. The control program for MAC, called IDCOM, is discussed in Section 8.1. Two control computation algorithms used in the simulations are described in Sections 8.2 and 8.3. The basic missile model is presented in Section 8.4, and modifications made to the roll axis are given in Section 8.5, along with step responses for the modified (compensated) missile. All simulation results will be presented in Section IX.

8.1 IDCOM Description

MAC is generally implemented by a computer program called IDCOM (for IDentification and COMmand) developed by ADERSA/GERBIOS in France. In order to simplify program modifications for our research, another version of IDCOM was written by us for simulating non-adaptive MAC applications. Our initial IDCOM is basically equivalent to the French version, with some modifications as described below.

A block diagram of the components of our IDCOM is shown in Figure 24. As indicated in the figure, when a new measurement is made, it is fed to two blocks of IDCOM and used to compute a reference trajectory and a "zero input" prediction of the future outputs over a short horizon (for optimization). The reference trajectory is a first-order exponential

115
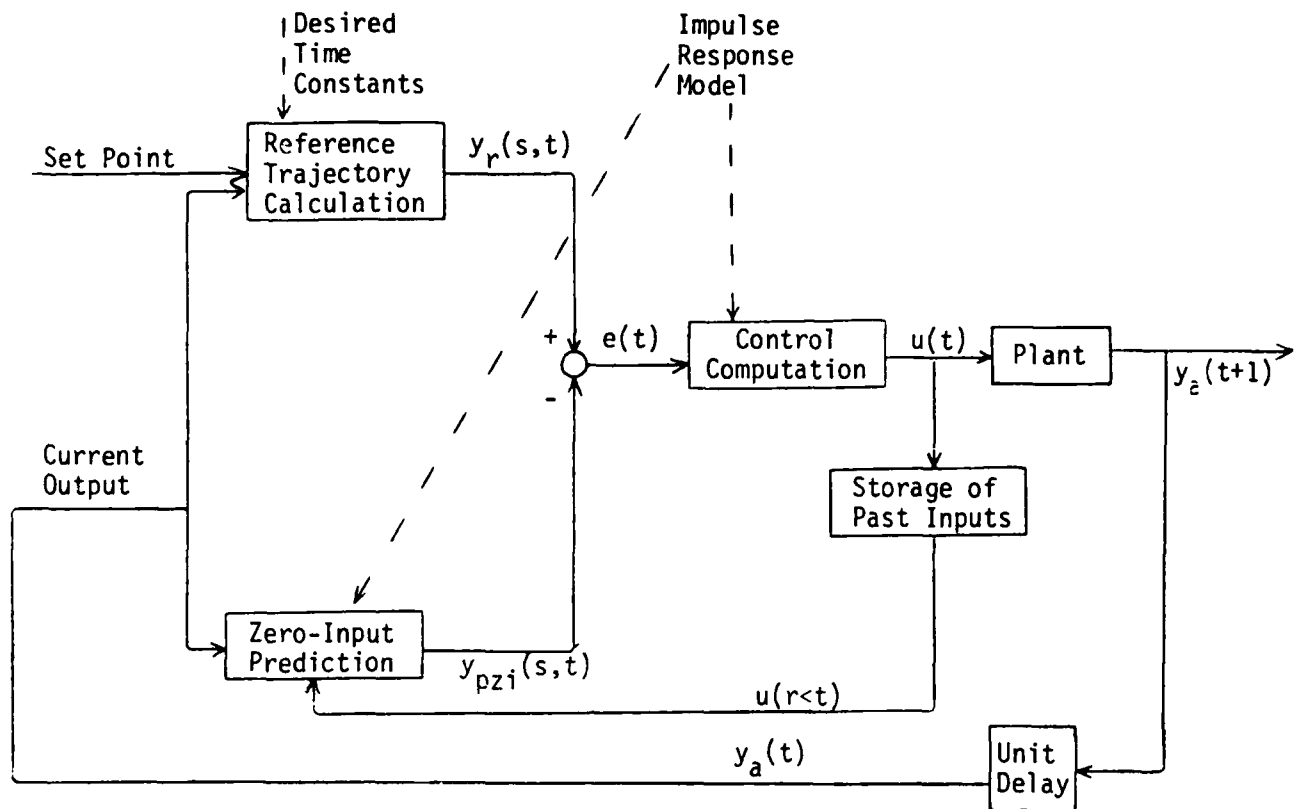
Figure 24.   IDCOM components.

drawn from the current (measured) output to a given set point. The de-
signer supplies a time constant (τ_i) for this exponential for each output
i. The "zero-input" prediction uses the past inputs, measurements, and
internal impulse-response model to predict the future outputs in the absence
of future control inputs.

These two trajectories (reference and zero-input prediction) are
differenced to obtain an error trajectory to be minimized by the future
controls. The control calculation block then performs this minimization
in one of several ways. Once an input sequence has been computed, the

116

first input is applied to the plant, and the cycle starts again after the next measurement.

For our simulations, the zero-input prediction was obtained by the "modified" (bias-compensated) estimator discussed in Section VII. The controls were calculated in one of two ways, as discussed below.

8.2 Basic Control Computation

Our first control computation block is essentially a simple inversion routine (in the absence of constraints) which finds inputs to zero the error between the reference and zero-input trajectories. To make the problem tractable, the error is only considered at a few points in the future, and the future controls are required to be constant over intervals ("blocks") between the output-matching points. For example, three input values u_1, u_2 and u_3 may be computed to define the next five future controls as:

$$u(1,t) = u_1$$
$$u(2,t) = u_2$$
$$u(3,t) = u_2$$
$$u(4,t) = u_3$$
$$u(5,t) = u_3$$

where $u(s,t)$ is the computed control at time t to be applied s steps in the future. This is shown pictorially in Figure 25.
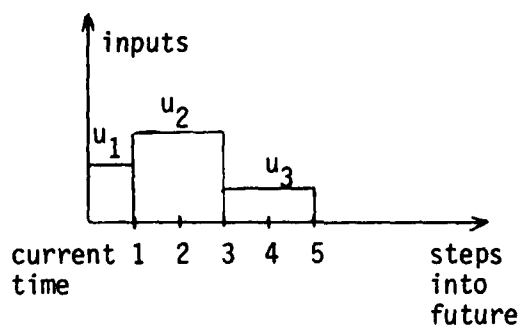
Figure 25. Input blocking.

For this case, the outputs to be controlled (the error to be reduced) occur at the endpoints of each block, i.e., 1, 3 and 5 steps in the future. The advantage of this blocking is that it permits reasonably long optimization horizons with low-dimensional required calculations. In this example, only *three numbers* are computed for five steps ahead. Since *the controls will* generally be recalculated one step ahead, this technique sacrifices little.

The optimization routine used in the basic algorithm tries to invert the system to find inputs which result in perfect output matching at the chosen points. If this is possible, then the first control causes the first output to be correct independently of the future inputs and outputs. If constraints are encountered, or if the computation time is shortened, however, the computed solution for one step ahead differs from the perfect control in a way that results in good performance at the future selected points.

8.3 Gradient Algorithm

In order to improve the control calculation for general systems, a gradient-projection algorithm was substituted for the basic (inversion) control computation. This new algorithm retains the input blocking discussed above (to reduce the dimension of the problem) but does not block the outputs. Instead, the new algorithm minimizes the sum of the squares of the output errors at each step in the future up to the end of the last input block. Thus, for the example above, the new algorithm would compute three control numbers to minimize five future errors.

The capability also exists in the new algorithm for adding input and output weighting matrices if desired. In general, we use only output weighting unless control cost is meaningful. Output weighting alone is very convenient, permitting easy tuning of the controller to achieve desired performance for each output. Finally, we note that, for only one block with endpoint one step ahead, the gradient algorithm and inversion routine are equivalent (without control weights).

8.4 Simulation Model

In order to verify the theoretical results obtained earlier and to demonstrate the behavior of MAC in an aerospace environment, a missile attitude control simulation was developed. A simple, three-axis attitude control model with independent pitch axis and coupled roll-yaw dynamics was chosen from AFIT (1978). The model represents a hypothetical air-to-air missile with asymmetric aerodynamic properties.

The model has six states, three inputs and three outputs, with

119

dynamics:

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

and

$$
A = \begin{bmatrix}
Z_w & 1 & 0 & 0 & 0 & 0 \\
M_\alpha & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & Y_\beta & \alpha_1 & -1 & \frac{g}{v}\cos\phi_1 \\
0 & 0 & L_\beta & 0 & 0 & 0 \\
0 & 0 & N_\beta & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
$$

$$
B = \begin{bmatrix}
0 & 0 & 0 \\
M_{\delta q} & 0 & 0 \\
0 & 0 & 0 \\
0 & L_{\delta p} & 0 \\
0 & 0 & N_{\delta r} \\
0 & 0 & 0
\end{bmatrix}
$$

$$
C = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

The states are:

$x_1 = \alpha$ angle of attack (rad)

$x_2 = q$ perturbed pitch rate (rad/s)

$x_3 = \beta$ sideslip angle (rad)

$x_4 = p$ perturbed roll rate (rad/s)

$x_5 = r$ perturbed yaw rate (rad/s)

$x_6 = \phi$ roll angle (rad)

with inputs

$u_1 =$ elevator angle (δ_q) (rad)

$u_2 =$ aileron angle (δ_p) (rad)

$u_3 =$ rudder angle (δ_r) (rad)

outputs

$y_1 =$ angle of attack (α) (rad)

$y_2 =$ sideslip angle (β) (rad)

$y_3 =$ roll angle (ϕ) (rad)

and parameters

$v =$ forward speed

$Z_w =$ dimensional variation of z-force with downward velocity, $\sec^{-1}$

$M_\alpha =$ dimensional variation of pitching moment with angle of attack, $\sec^{-2}$

$Y_\beta =$ dimensional variation of y-force with sideslip angle, $\sec^{-1}$

L_β = dimensional variation of rolling moment with sideslip angle, $\sec^{-2}$

N_β = dimensional variation of yawing moment with sideslip angle, $\sec^{-2}$

$M_{\delta q}$ = dimensional variation of pitching moment with pitch control surface deflection, $\sec^{-2}$

$L_{\delta p}$ = dimensional variation of rolling moment with roll control surface deflection, $\sec^{-2}$

$N_{\delta r}$ = dimensional variation of yawing moment with yaw control surface deflection, $\sec^{-2}$

g = acceleration of gravity

ϕ_1 = equilibrium roll angle

Two flight conditions were chosen for study. The first has the missile at Mach 2 at 20,000 ft, and weighing 239.5 lb. The pitch angle (equilibrium) is 9° and sideslip is 0°. The second flight condition is Mach 4 at 70,000 ft, same weight, and equilibrium pitch and sideslip of 14° and 11° respectively.

The parameters for flight condition 1 were:

$$Z_w = -1.4868$$
$$M_\alpha = -149.93$$
$$M_{\delta q} = -281.11$$
$$Y_\beta = -.91237$$
$$L_\beta = -1559.2$$
$$L_{\delta p} = 8770.6$$
$$N_\beta = 290.48$$
$$N_{\delta r} = 281.11$$

The parameters for flight condition 2 were:

$$Z_w = -.27877$$

$$M_\alpha = -64.928$$

$$M_{\delta q} = -43.285$$

$$Y_\beta = -.25089$$

$$L_\beta - 1650.5$$

$$L_{\delta p} = 1875.7$$

$$N_\beta = 50.499$$

$$N_{\delta r} = 43.285$$

The models are considered acceptable for variations of 3° in α and β.

8.5 Model Modifications

An early *impulse response* analysis of these dynamics indicated a very severe roll instability. Since IDCOM works best with finite impulse responses, we chose to add roll angle and rate feedback to the aileron command, thus creating a compensated system for IDCOM to control. This changed the fourth row of the A matrix to be:

$$[0 \quad 0 \quad L_\beta \quad -L_{\delta p}G_\phi \quad 0 \quad -L_{\delta p}G_\phi]$$

Due to the roll-yaw decoupling from the pitch axis, this change did not affect the angle-of-attack dynamics at all.

In the above row, $L_{\delta p}$ varies with flight condition, but G_ϕ, the compensator gain, was fixed to be ½ in our tests. AFIT (1978) presented values of G_ϕ of 0.127 at flight condition 1 and 1.056 at flight condition 2,

123

but we elected to use a simple, fixed gain compensator to merely stabilize the plant. IDCOM would be used to get the required performance, and thus an "optimal" G_ϕ should not be needed.

Step responses of this compensated plant to control inputs are shown in Figures 26, 27 and 28 for flight condition 1 and Figures 29, 30 and 31 for flight condition 2. A sample rate of 10 Hz (10 samples per second) was used for these plots, with linear interpolation (by the plotting routine) filling in between data points. As shown in the plots, the pitch axis dynamics are quite oscillatory for both flight conditions. An analysis of the decoupled pitch dynamics revealed that flight condition 1 has a natural frequency of 12.24 r/s (1.95 Hz) and a damping ratio (ζ) of 0.061, while condition 2 has a frequency of 8.06 r/s (1.28 Hz) and a damping ratio of 0.017.

8.6 Summary

This section has described the application of MAC to a missile attitude control simulation. Two versions of the control algorithm (IDCOM) were described, and their differences discussed. The missile model was presented, and changes for our studies were noted. The next section will present the results of these simulations.
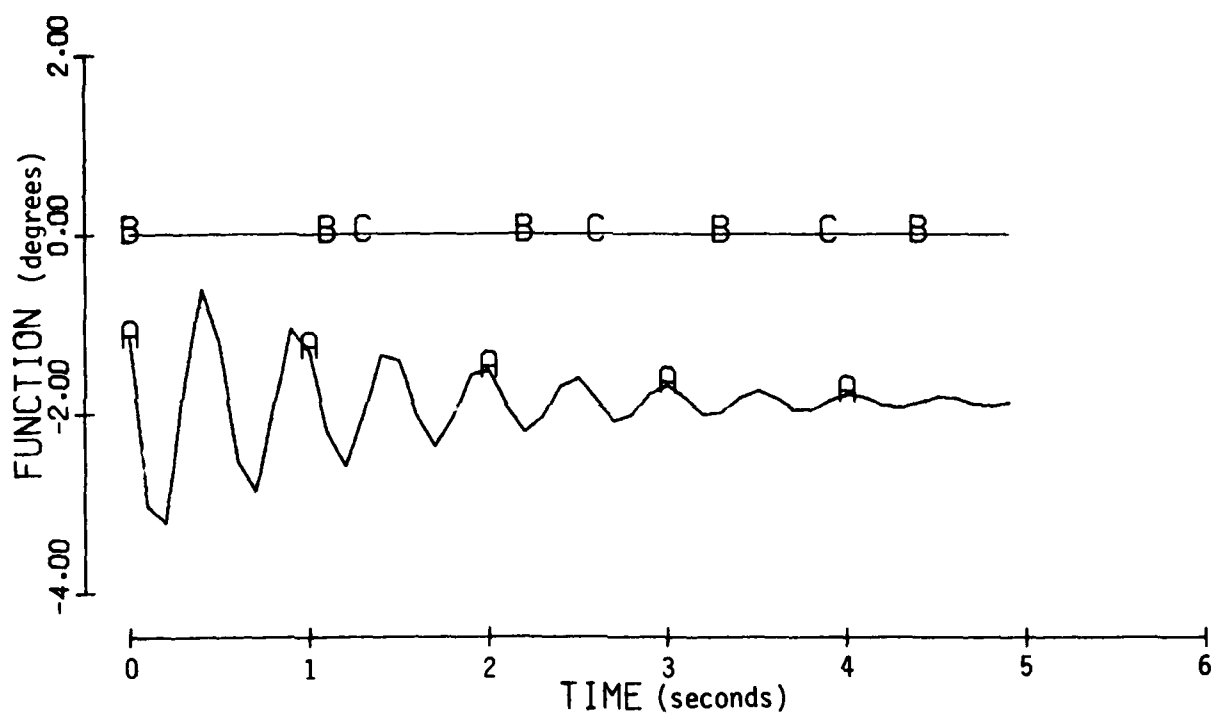
Figure 26. Step response to elevator input, flight condition 1.

125

A = angle of attack
B = sideslip angle
C = roll angle
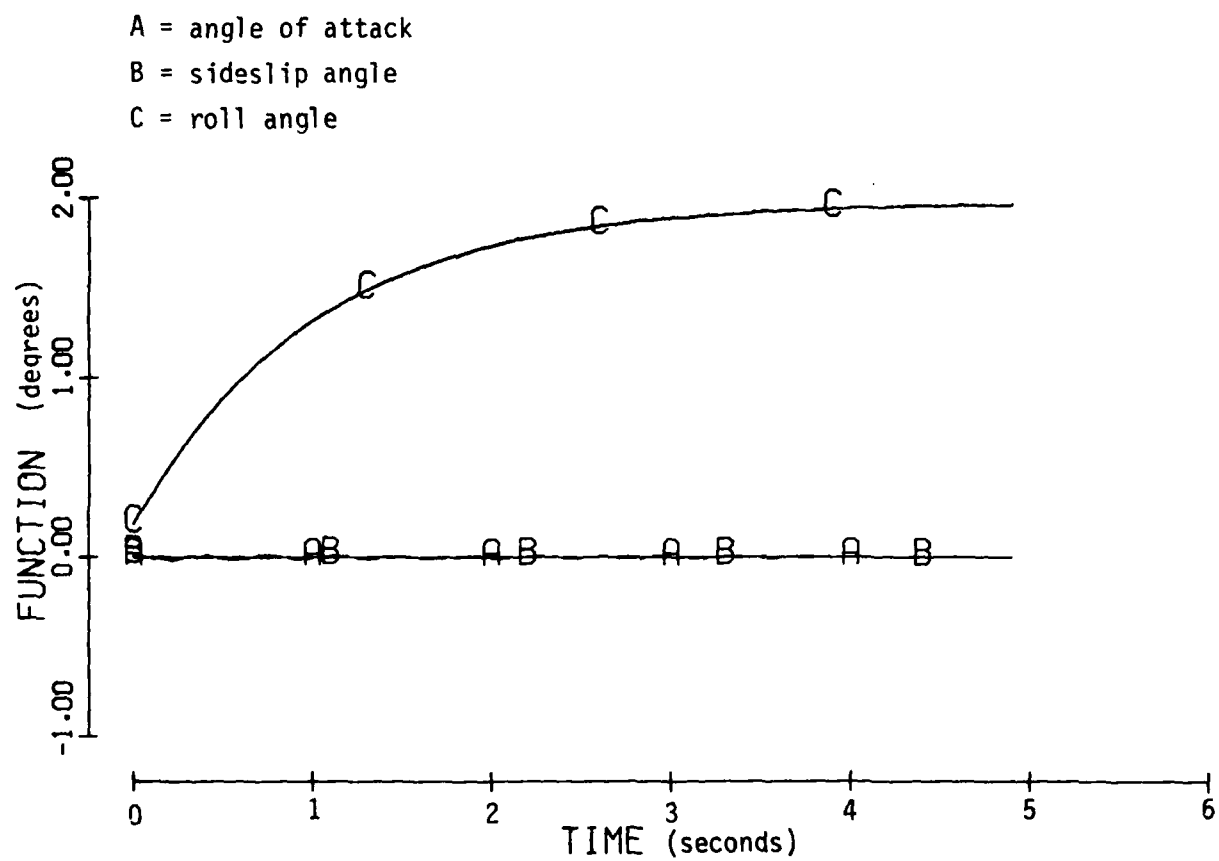
Figure 27. Step response to aileron input, flight condition 1.
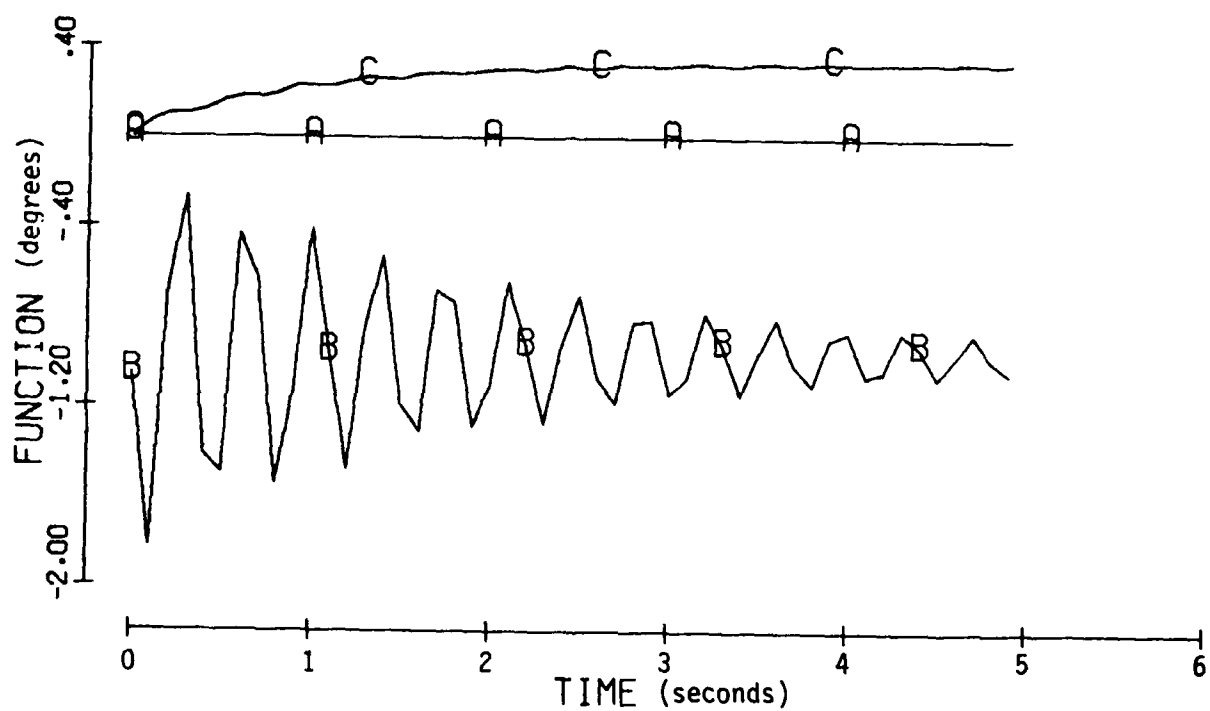
Figure 28. Step response to rudder input, flight condition 1.
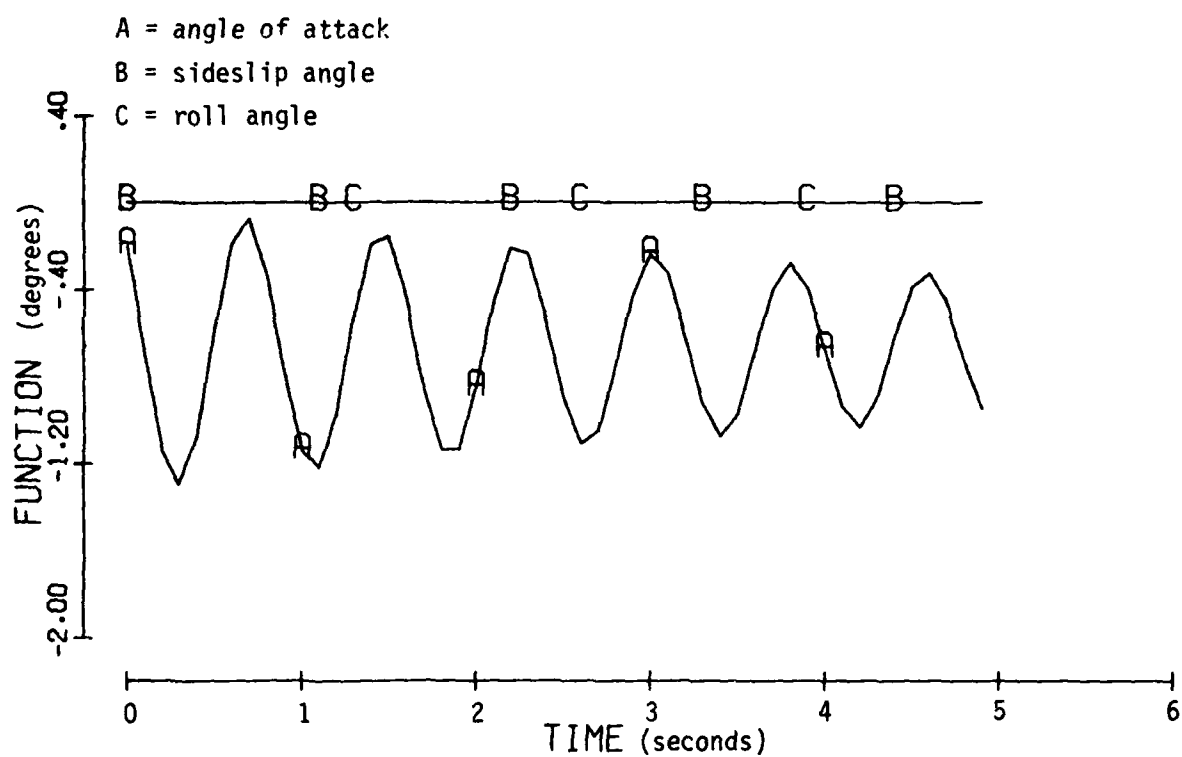
Figure 29. Step response to elevator input, flight condition 2.

128

A = angle of attack
B = sideslip angle
C = roll angle
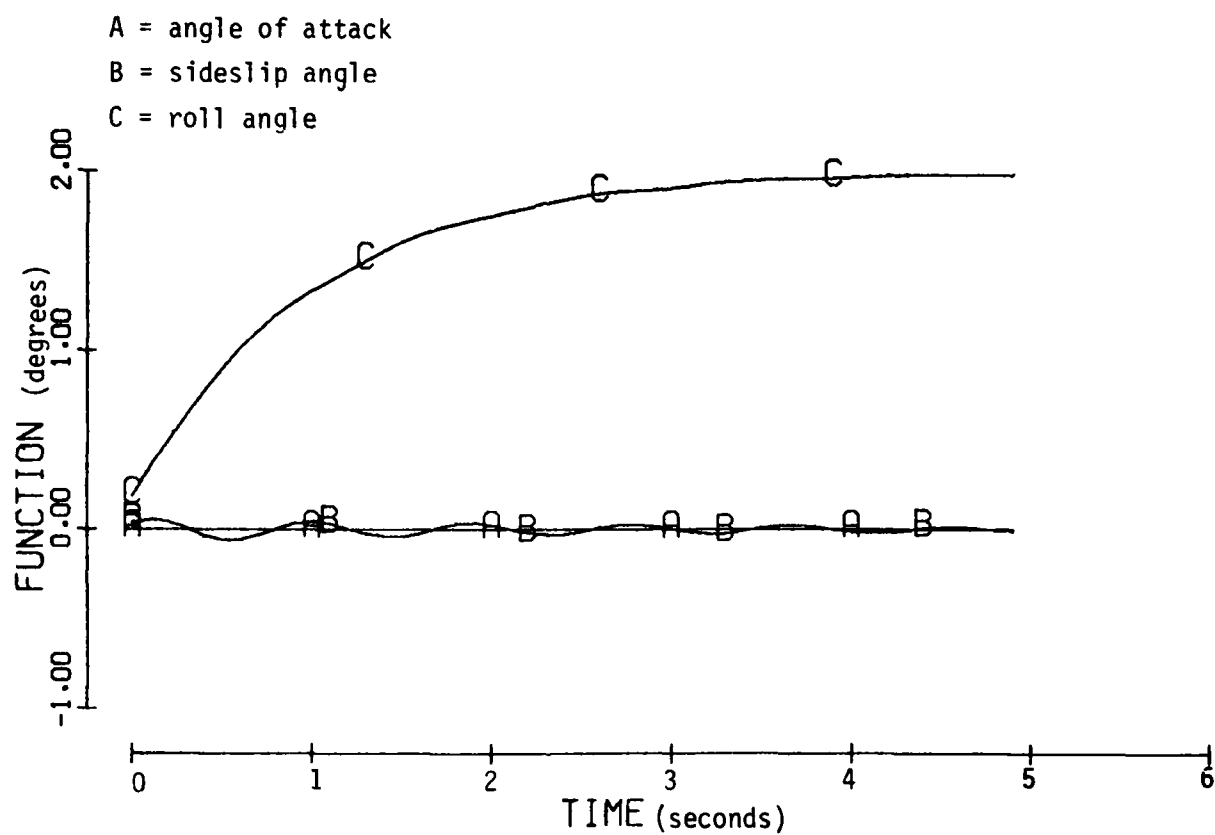
Figure 30. Step response to aileron input, flight condition 2.

129
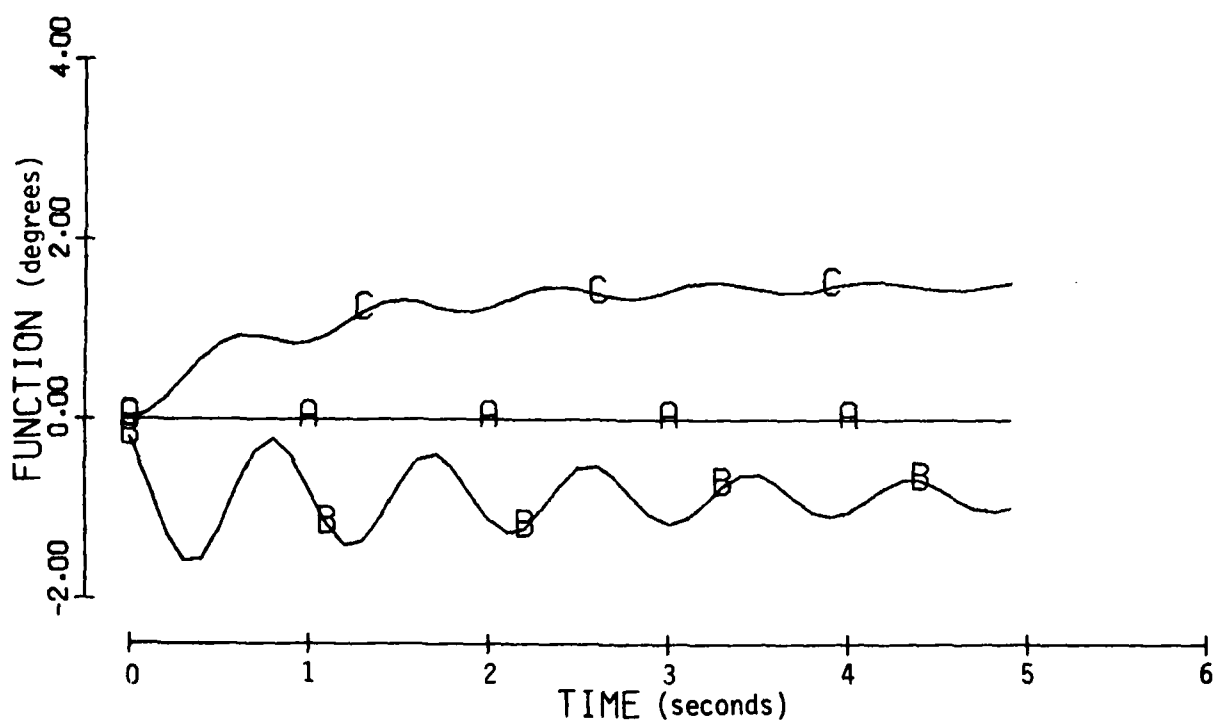
Figure 31.   Step response to rudder input, flight condition 2.

SECTION IX

SIMULATION RESULTS

Overview

This section presents the results of MAC applied to a missile attitude control simulation as described in Section VIII. The first section (9.1) lists the simulation parameters and baseline conditions for the majority of the runs. Section 9.2 then shows the performance of both the basic (inversion) algorithm and the modified (gradient) algorithm for the baseline conditions and some slightly different parameter values. Changes in the reference trajectory time constant are shown in Section 9.3, and the effects of input rate constraints demonstrated in Section 9.4. Section 9.5 shows the behavior of the algorithms when measurement noise is present, and Section 9.6 demonstrates robustness limits for cases where the internal model and real missile are not the same.

9.1 Simulation Parameters

The following sections will present a number of plots showing control responses as several different parameters and conditions are varied. In order to facilitate comparison between related plots, the scales have been kept constant, if possible, within each series of runs.

Unless otherwise noted, the following conditions existed in the simulations:

•The sample time (Δ) was 0.1 second.

- The controls were computed for three blocks ending at one, three, and five steps in the future.

- The reference trajectory time constant (τ) was 0.1 seconds (all axes).

- No input constraints were imposed.

- The set points were changed at 0.4 seconds from 0° to 15° for angle of attack (α) and 10° for sideslip. The roll set point remained at 0°.

- For the gradient algorithm, the output weights w_i were all equal (1); no input weights were used.

- Flight condition 1 was used.

The sampling rate of 10 Hz corresponds to a Nyquist frequency of 5 Hz, and thus all plant dynamics should be slower than 4 Hz. We expected the closed loop plant to typically be controlled with a 0.1 second reference trajectory time constant, which would qualitatively match the performance shown in AFIT (1978). This time constant is equal to a 1.59 Hz natural frequency, which is sufficiently below the 5 Hz limit for these simple simulations.

For the simulation, the impulse responses for IDCOM's internal model were obtained numerically from the state space parameters. The simulation itself (i.e., the plant) used a discrete-time state model obtained from solving for the transition matrix for the state. The impulse responses and transition matrix calculations were carried out with a 0.001 second step size for accuracy. This method insured that accurate dynamic behavior would be seen at the sample points, although no attempt was made in our tests to examine the response between samples.

132

We note that several responses are outside of the linear range for the model, and thus should not be considered as absolute results. Rather, the inputs and outputs should all be scaled down (to less than 3°) if absolute numbers are desired. Since all of the operations (except for the constraint case) are linear, this presents no problems.

We also note that the low pitch damping ratio of the missile at each flight condition presents a difficult control task for any digital controller. An actual missile implementation would benefit from an analog stabilization loop inside of the IDCOM control loop (as was modeled for the roll axis and discussed in Section VIII). We did not use this internal control structure for angle of attack, however, since we wanted to explore the operation and limitations of different IDCOM techniques in this environment.

9.2 Initial Control Tests

This series of tests demonstrates the basic control behavior of the two versions of IDCOM tested. The first figure (Figure 32)[14] shows the basic (inversion) algorithm for the baseline conditions described above. Figure 33 shows the same controller but only looking one step ahead (i.e., one endpoint, one step ahead). We see that without constraints, the results are nearly identical. The large oscillations in inputs are due to the inversion of an oscillatory system (the missile). The controlled outputs are very good, but the control behavior appears undesirable.

---

[14]All figures for Section IX are grouped at the end of the section, beginning on page 141.

133

The next five plots show the modified, gradient algorithm. Figure 34 shows the baseline conditions, and Figure 35 shows the response to a 10° roll command with 0° sideslip. Note the _much_ smoother control sequence compared to the inversion routine (Figure 32). Figure 36 shows the gradient algorithm only looking one step ahead, and thus converging to the inversion technique. Figure 37 shows the improvement of just considering two endpoints (one and five steps ahead), and Figure 38 considers three endpoints (one, three, and 10 steps ahead), using a longer optimization horizon. This results in slightly slower initial response but smoother controls than Figure 34.

9.3 Reference Trajectory Changes

These tests show the effect of reference trajectory time constant on speed of response. The gradient algorithm was used for this series and results should be compared to Figure 34. Figure 39 shows the response for a time constant of .5 second (from .1 seconds in Figure 34), while Figure 40 speeds the reference to 0.05 seconds. For comparison, the fast time constant (.05) with a one-step look ahead controller (gradient here, but equivalent to the inverse) is shown in Figure 41.

9.4 Input Rate Constraints

This series of runs shows the effect of adding rate constraints to the control inputs. IDCOM directly considers magnitude and rate constraints in its control calculation so that improved compensation is possible (i.e., other controls may be adjusted) when one input is limited. For the

missile control simulation, magnitude limits did not seem important, since most control surfaces had large travel available. Rate limits (i.e., derivative of control surface motion) did seem realistic, however, and we attempted to investigate their effects on performance.

We recall that the baseline tests showed large swings in control inputs were needed for the inversion algorithm to achieve its excellent tracking. These input rates, although alarming, barely exceed a 250°/second rate limit suggested by the Air Force for this simulation. Figure 42 shows the response of the inversion algorithm (one step look ahead) to a rate limit of 250°/second. The performance is generally indistinguishable from that of the baseline test, Figure 32. A very strict 50°/second rate limit is imposed in Figure 43, and this time the output does not perfectly match the reference trajectory but the control swings are nearly eliminated.

9.5 Effects of Noise

This series of tests demonstrates the behavior of IDCOM in the presence of white, Gaussian measurement noise. The gradient controller was used for these runs, although the first example (Figure 44) uses only one optimization point and is the equivalent of the inversion technique. The controller was used in a regulator mode, with no set point step commands, and the plot scale was reduced to show the output noise. The plots are actually of the measured output including the true plant output plus measurement noise. They thus show slightly more motion than the true plant output alone exhibited. The noise was chosen to have a variance of 0.25, applied at the sample time, 0.1 seconds apart.

Figure 44 shows the outputs for the inversion-equivalent case. The actual output noise variances were 0.1348, 0.0922, and 0.1001 for angle of attack, sideslip, and roll respectively. Figure 45 shows the gradient algorithm looking three steps ahead (as in the baseline case). The output noise variances for this case are 0.1016, 0.0387, and 0.0290 for angles of attack, sideslip and roll, respectively.

Although it is dangerous to draw conclusions from too small a random sample, the gradient algorithm appears to retain its smooth control properties (compared to the inversion technique) with little or no penalty in output noise variance. The much more easily analyzed inversion routine may, therefore, provide a rough estimate of noise behavior for the gradient technique.

9.6 Model Mismatch

This section demonstrates the behavior of the IDCOM algorithms when the internal (to IDCOM) impulse response model is not the same as the state space plant model used to simulate the missile. Two different types of mismatch are shown below: a simple gain error as discussed in Section 3.1, and a flight condition mismatch (e.g., internal model of flight condition 1 with plant from flight condition 2) as treated in Appendix B.

9.6.1 Gain Mismatch

As discussed in Section 3.1, when the internal model impulse response is equal to a constant gain $(1/q)$ times the real impulse response, the controlled system is stable if

$$0 < q < \frac{2}{1-\alpha} \qquad (9.1)$$

where

$$\alpha = e^{-\Delta/\tau}$$

for sample time Δ and reference trajectory time constant τ. This assumes that a stable system is being controlled, and that the inversion algorithm is being used.

To demonstrate this result we used the gradient controller with one endpoint one step ahead (equivalent to the inversion routine) and set point step commands of 3° (angle of attack), 2° (sideslip) and 0° (roll). These set points have the same relative value as the baseline conditions and permitted use of the same plot scaling.

For the normal sample time ($\Delta = 0.1$ seconds) and time constant ($\tau = 0.1$ seconds), equation (9.1) indicates instability for

$$q > 3.164$$

Figure 46 shows the case where $q = 3$, and indeed the system is barely stable. Increasing q to 4 resulted in instability (not shown).

As indicated by equation (9.1) for any given q, robustness can be improved by increasing the reference trajectory time constant. For $q = 4$,

$$\tau > 0.144 \text{ seconds}$$

should produce stability. Indeed, a case with $\tau = 0.15$ is shown in Figure 47, and the system is again stable, but not by much. Increasing

137

τ to 0.2, as shown in Figure 48 ($q=4$), made the system much more stable (less oscillatory).

Although no theoretical results exist yet for the gradient routine, its generally smoother control was expected to result in greater robustness than the inversion algorithm. This proved to be true in our limited testing. Figure 49 shows the gradient algorithm with three endpoints (baseline conditions), $q=3$, and $\tau=0.1$ seconds as in Figure 46. The gradient algorithm's behavior is clearly superior. Indeed, for $q=4$, $\tau=0.1$ seconds, when the inversion case was unstable, the gradient algorithm gave reasonably good performance, as shown in Figure 50.

These tests seem to both confirm the theory of Section 3.1 and imply that the robustness bounds for the simple inversion are conservative limits for more sophisticated optimization algorithms.

9.6.2   Flight Condition Mismatch

One of the most attractive possibilities of robust control of aircraft and missiles is the potential for reducing the amount of gain scheduling required as flight conditions change. Robustness analysis for this case can be quite tedious, as indicated in Appendix B, and one often must simulate and examine controllers to determine the best structure and gains. This section presents a very limited demonstration of the potential of IDCOM in these areas.

The two flight conditions chosen (see Section 8.4) represent widely different systems, each difficult to control. Figures 34 and 36 earlier showed the performance of IDCOM (gradient version) for three and one endpoint optimization (respectively) at flight condition 1. Similar

baseline plots for flight condition 2 are given in Figures 51 (three endpoints) and 52 (one endpoint), but with step commands of 3° (angle of attack), 2° (sideslip) and 0° (roll). Figure 52 shows the near instability of the inverse control technique for the highly undamped pitch dynamics at this flight condition. Again, the gradient technique (Figure 51) is much better. In all of these runs, the internal model was correctly identified (off-line) for each flight condition.

We next tried mixing the two internal models and actual flight conditions, to see if one control model would work at the two extreme conditions. We simulated both gradient and inversion techniques, and neither was stable with the mismatched model for a reference τ of 0.1 seconds. We then tried slowing the reference trajectory down, and were able to stabilize the systems, with the consequent sluggish performance.

Figure 53 shows the inversion (actually gradient with one endpoint) algorithm ($\tau = 1$ second) at flight condition 2 and an internal model from flight condition 1. The set points were the baseline 15°, 10°, and 0°. Figure 54 shows the same case for a three endpoint, gradient algorithm. The responses are very similar, with less control oscillation in the gradient case.

Figure 55 shows the inversion algorithm for the reverse case (model from condition 2 and missile simulated at condition 1) with lower set points (3°, 2°, and 0°). The missile immediately approaches the set point because of the unexpectedly large response (compared to the control calculation model), even though the time constant is still 1 second (reduced for stability). Figure 56 shows the gradient (three point) algorithm for this case with the familiar improvement in control smoothness.

139

9.7 Summary

This section has presented the simulation results from applying MAC to a missile attitude control model. Some of the theoretical results of the earlier analysis sections have been demonstrated for a simple algorithm which approaches the idealized MAC considered in Section III. Results were also presented for an improved controller currently under development.
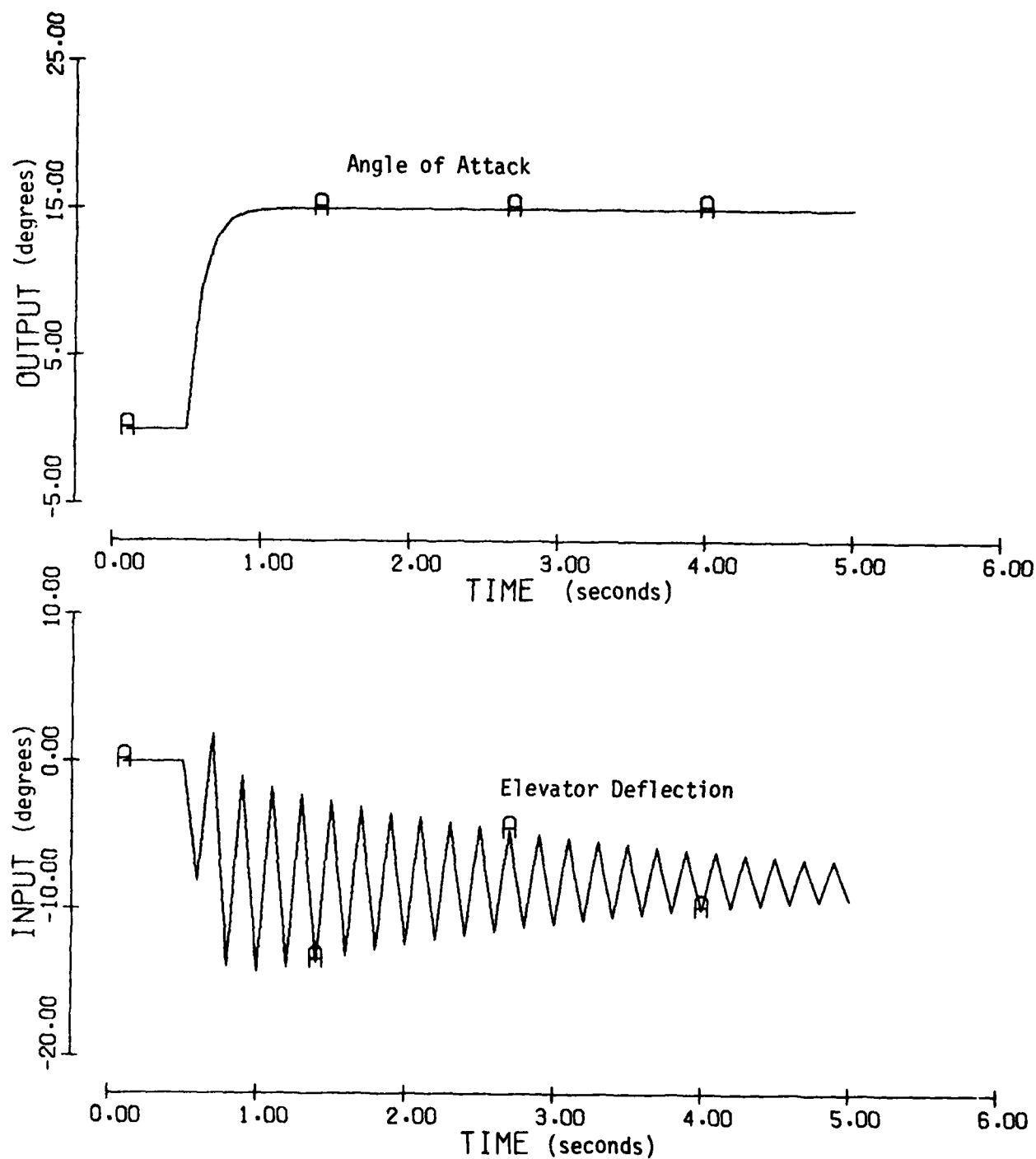
Figure 32a. Inversion algorithm at baseline conditions.

141
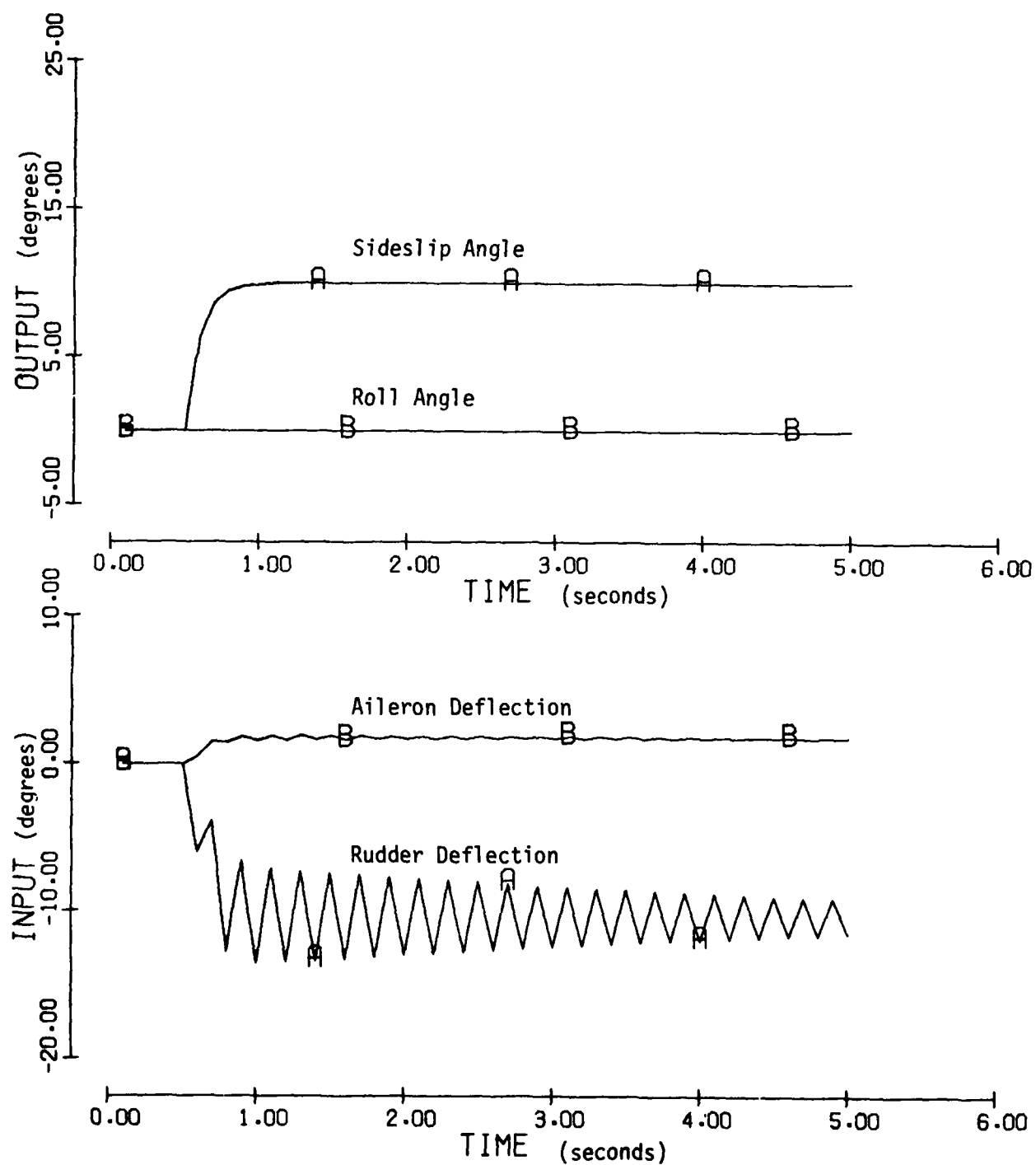
Figure 32b. Inversion algorithm at baseline conditions.

142
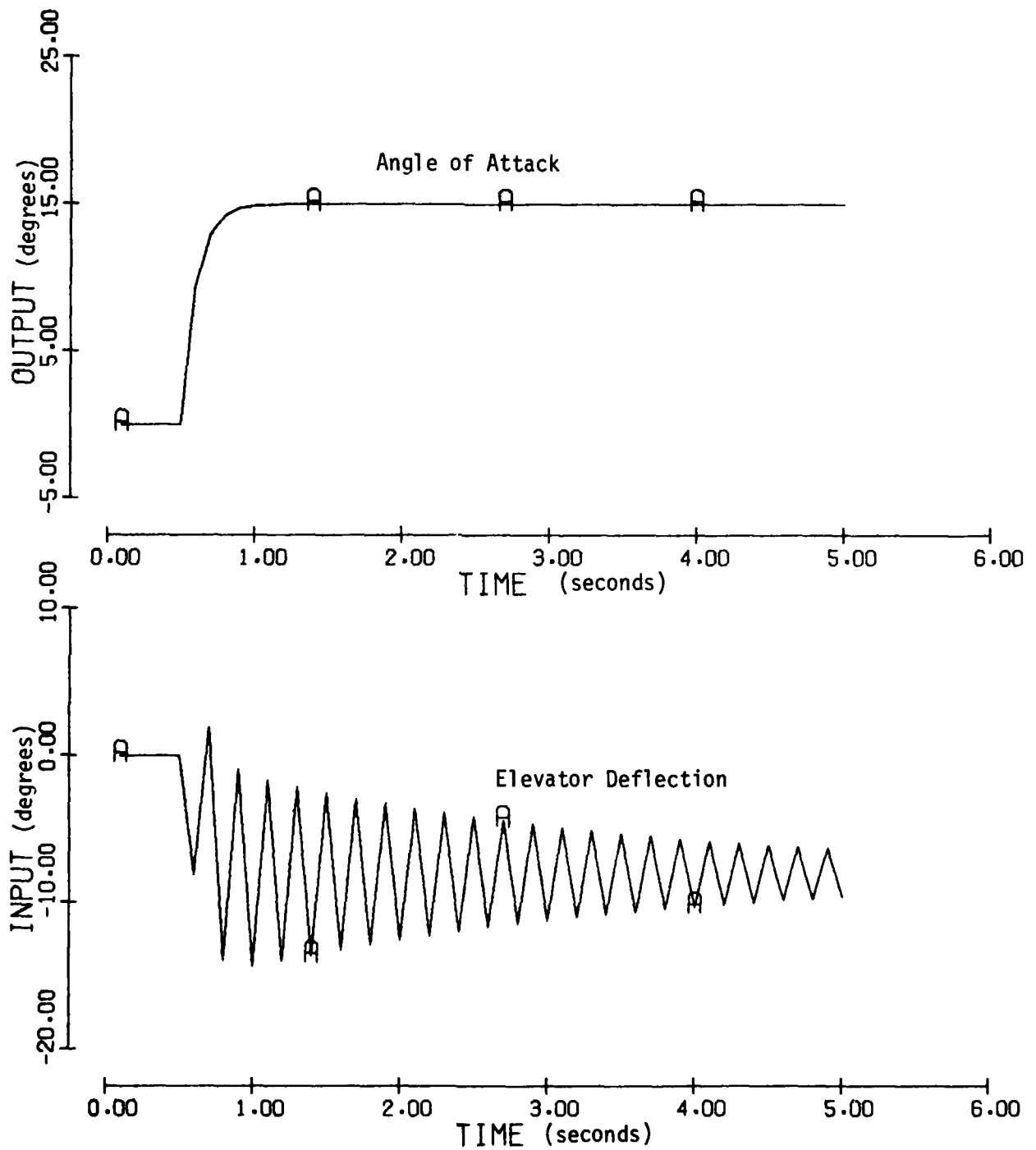
Figure 33a. Inversion algorithm with one endpoint.

143
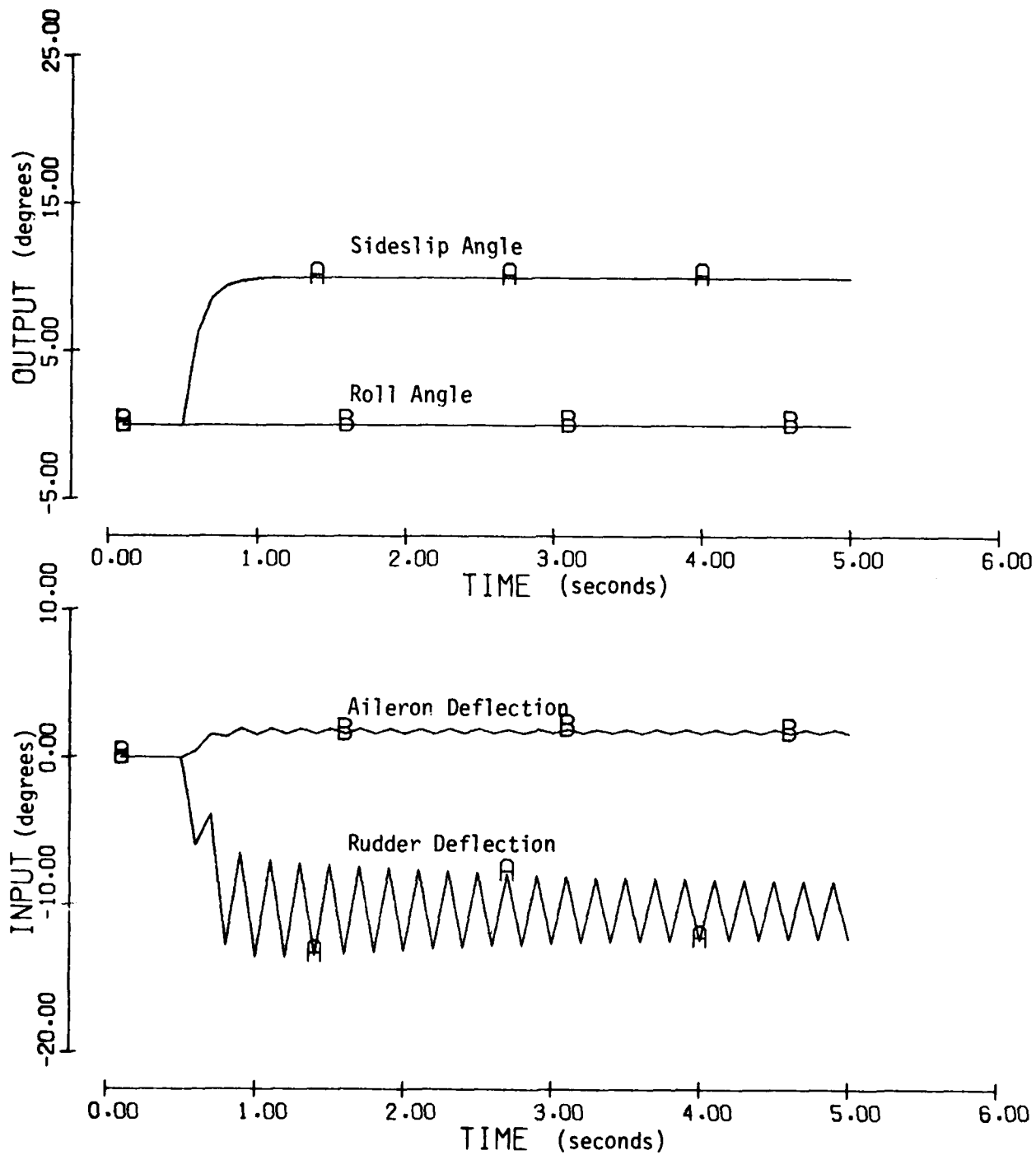
Figure 33b. Inversion algorithm with one endpoint.

144
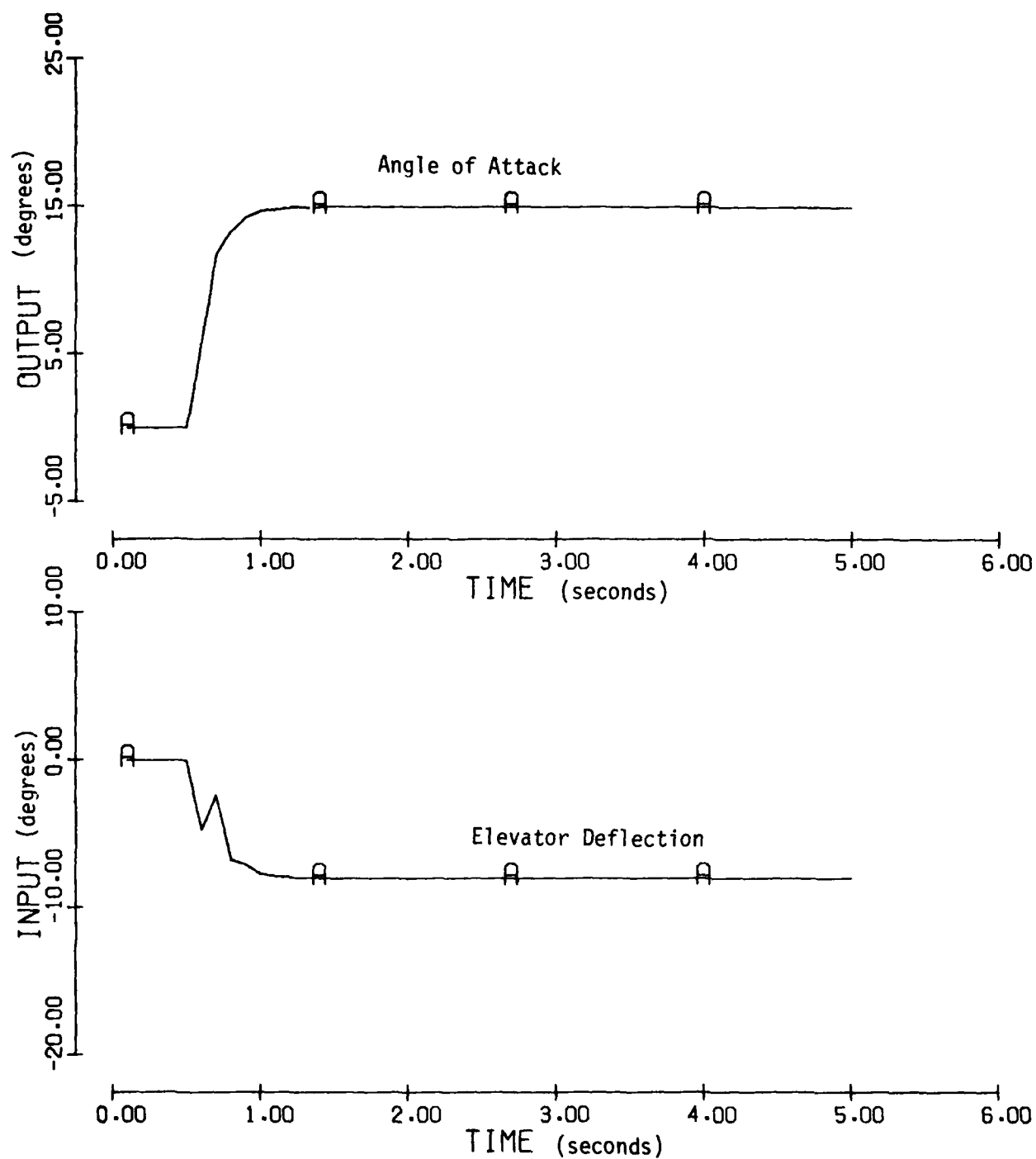
Figure 34a.   Gradient algorithm at baseline conditions.

145
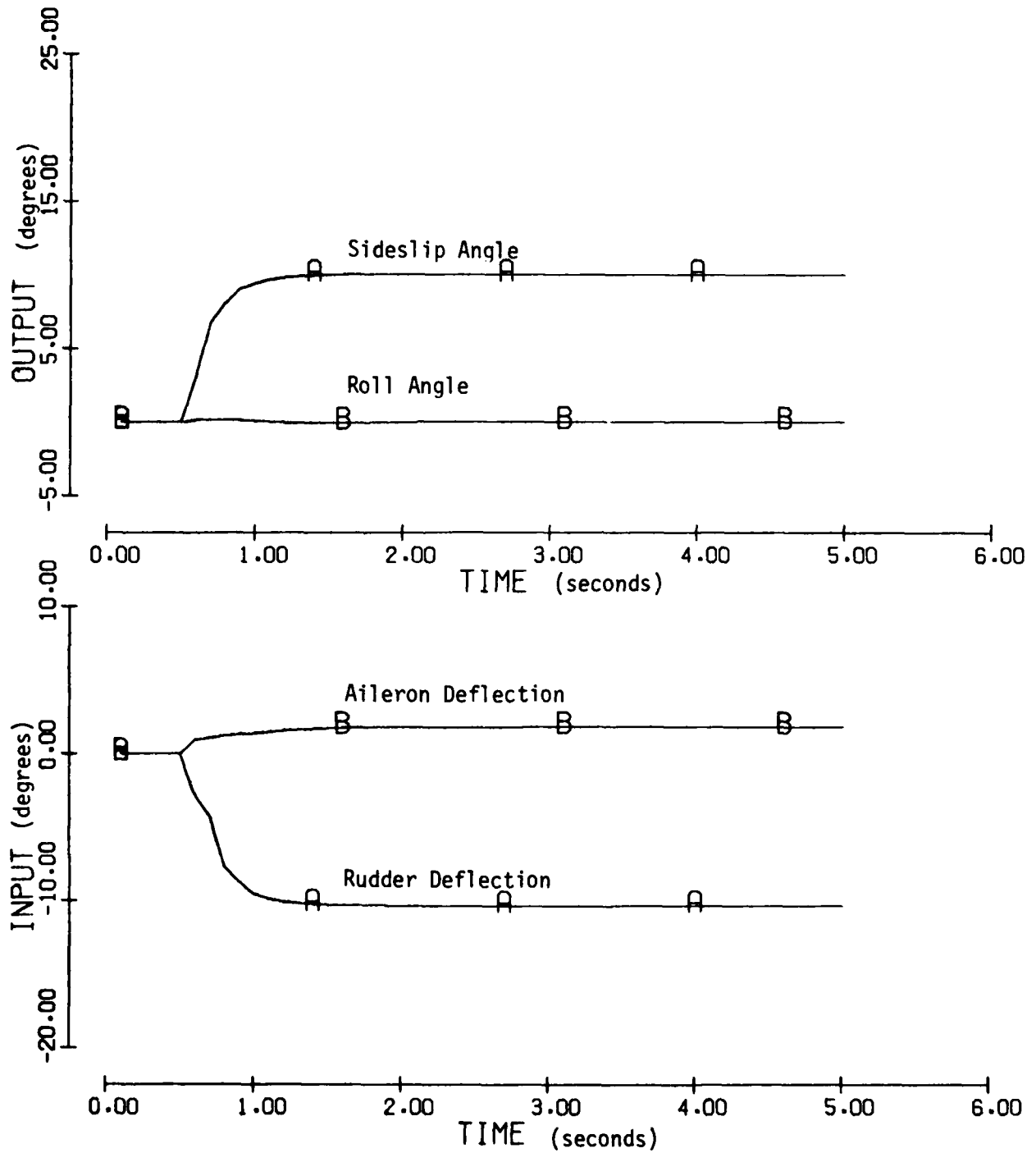
Figure 34b. Gradient algorithm at baseline conditions.

146
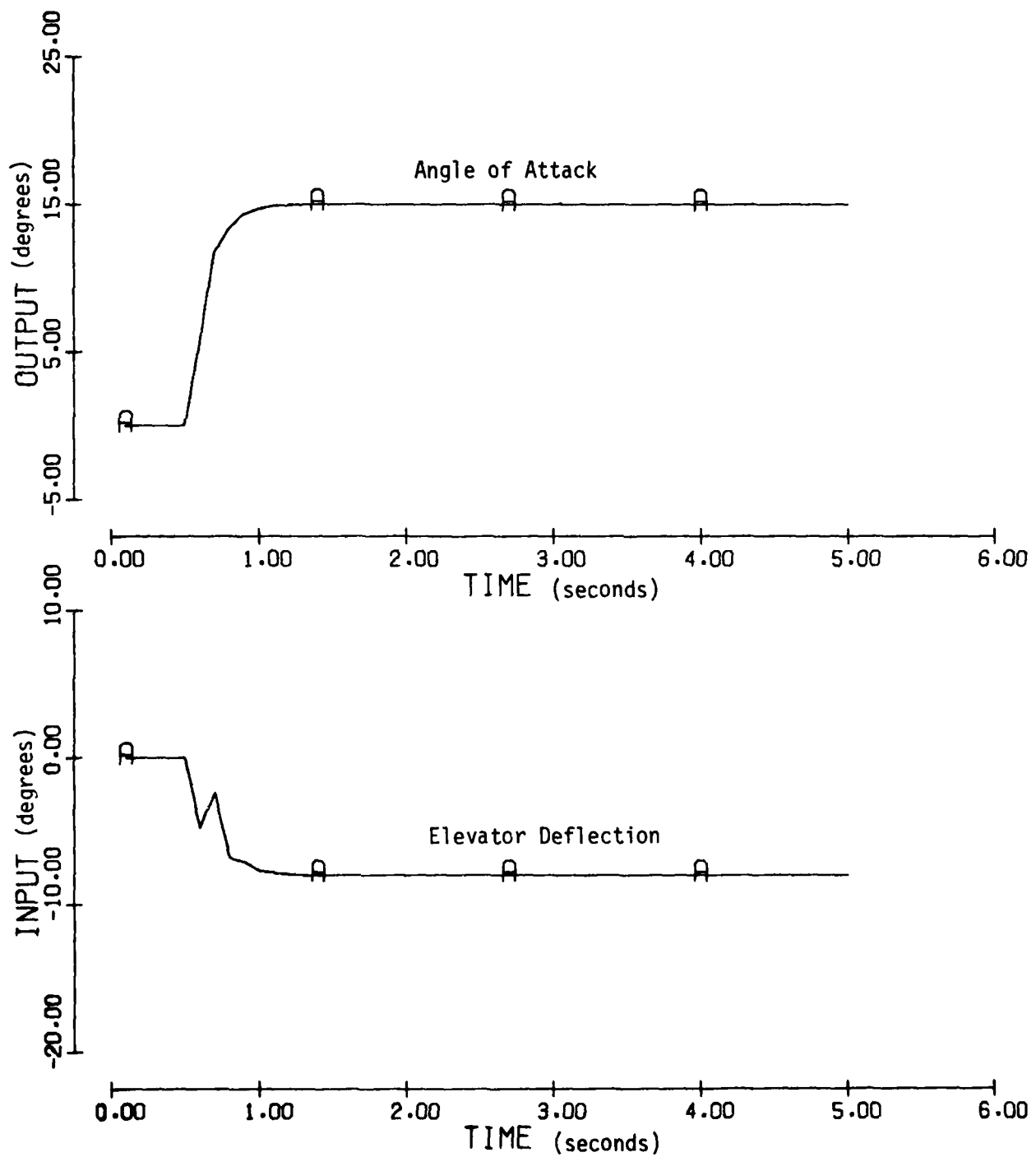
Figure 35a.   Gradient algorithm at baseline conditions, roll command.

147
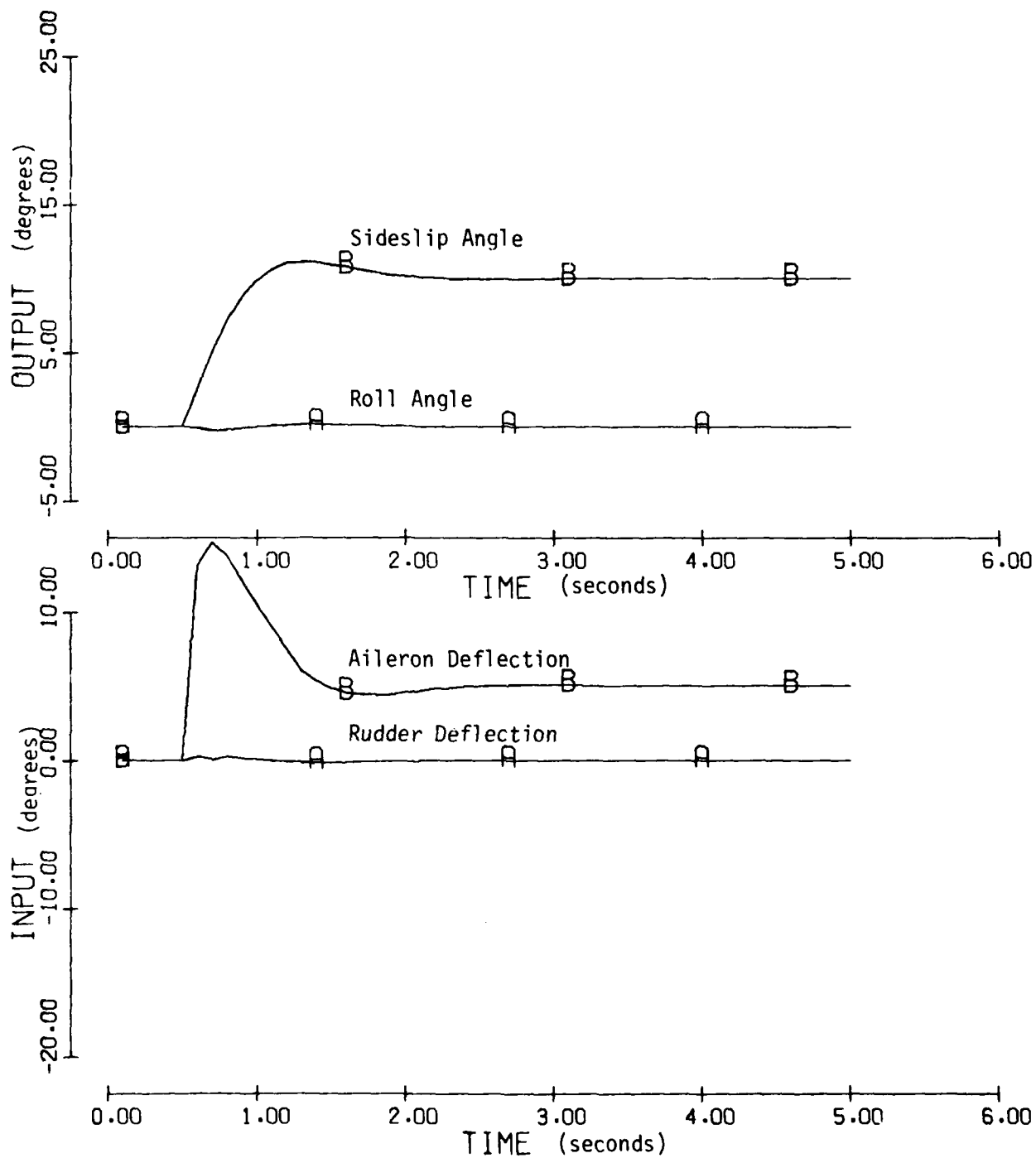
Figure 35b. Gradient algorithm at baseline conditions, roll command.

148
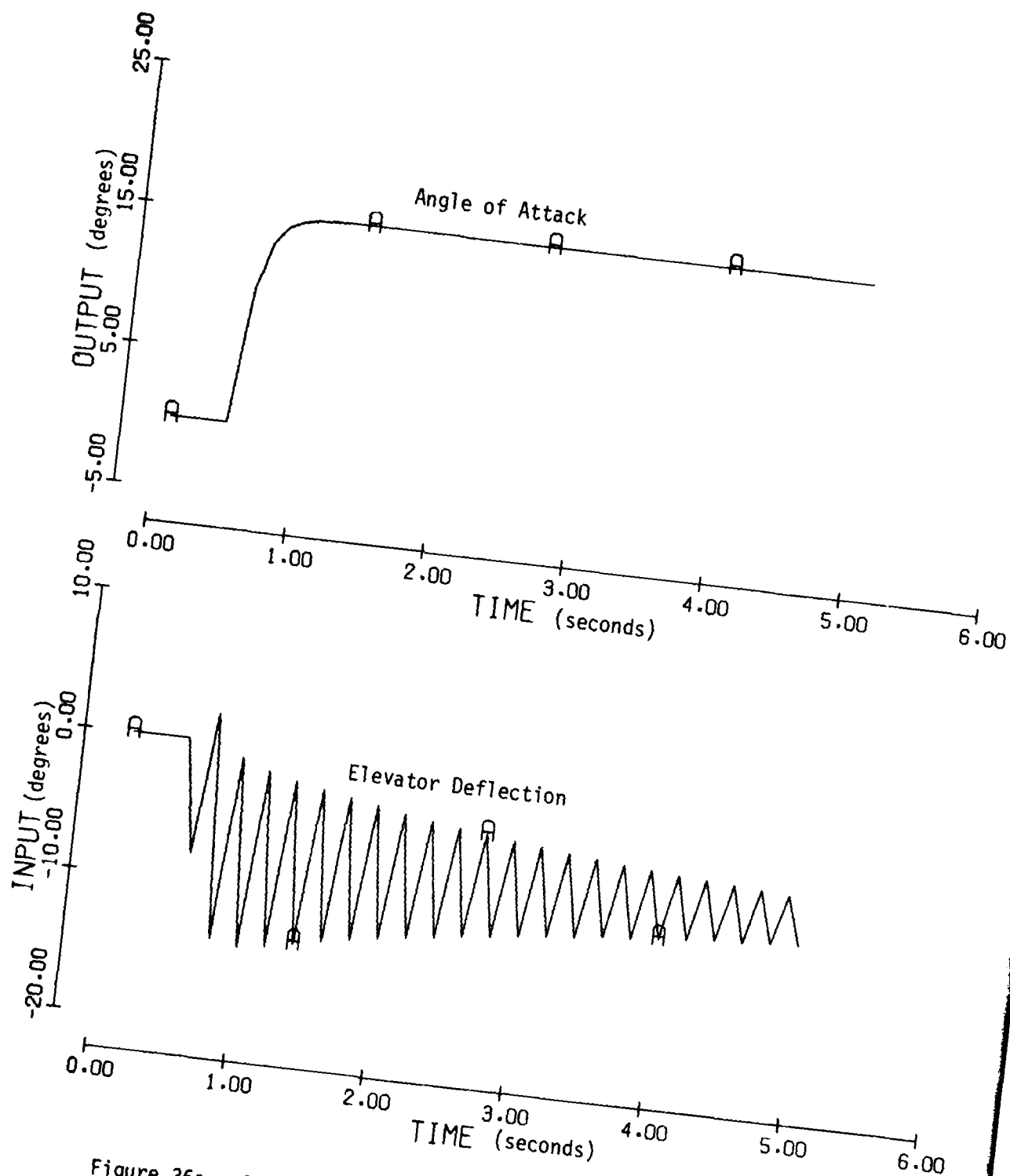
Figure 36a. Gradient algorithm with one endpoint (becomes inversion algorithm, cf. Figure 33).

149
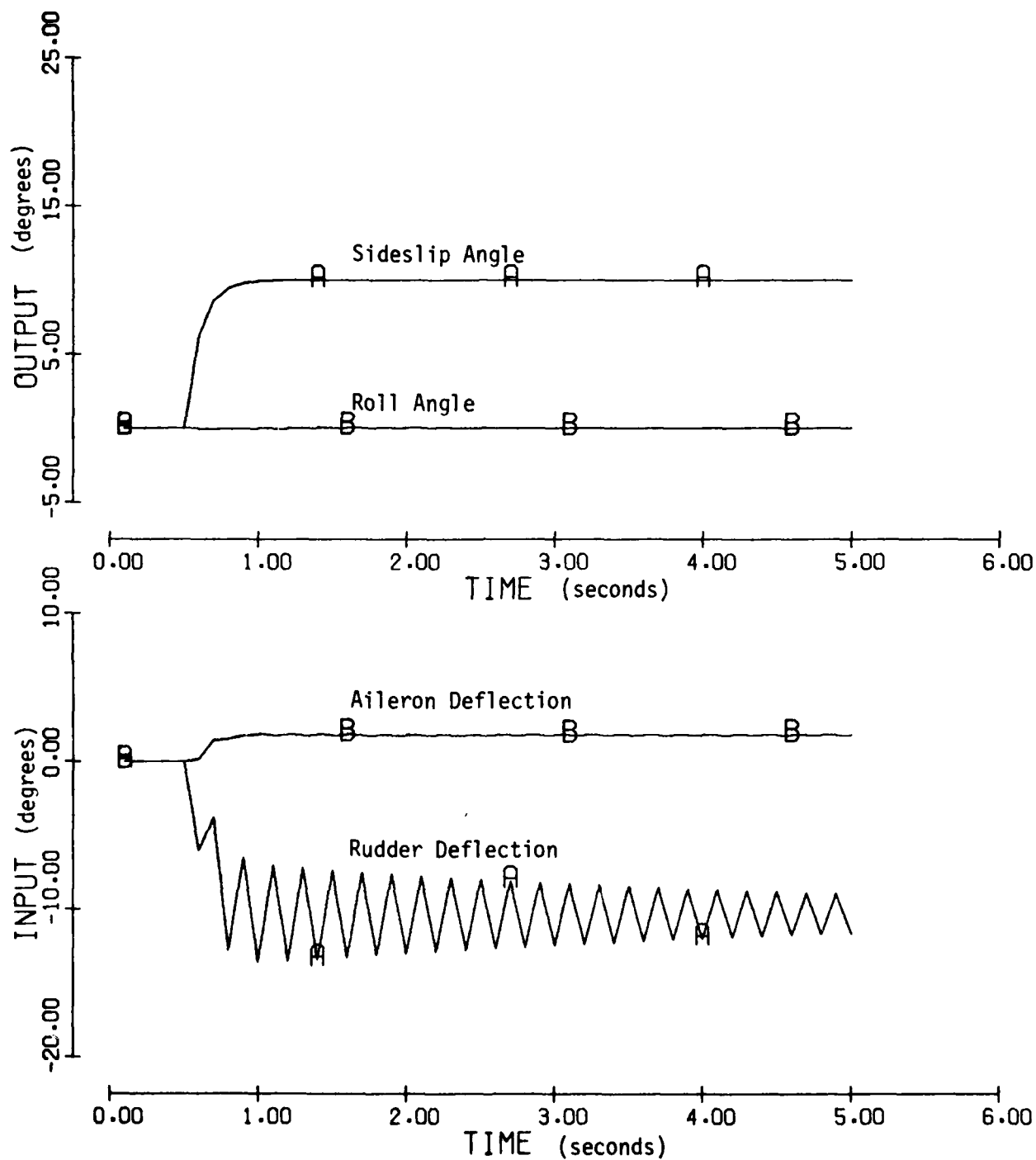
Figure 36b.   Gradient algorithm with one endpoint (becomes
inversion algorithm, cf. Figure 33).

150
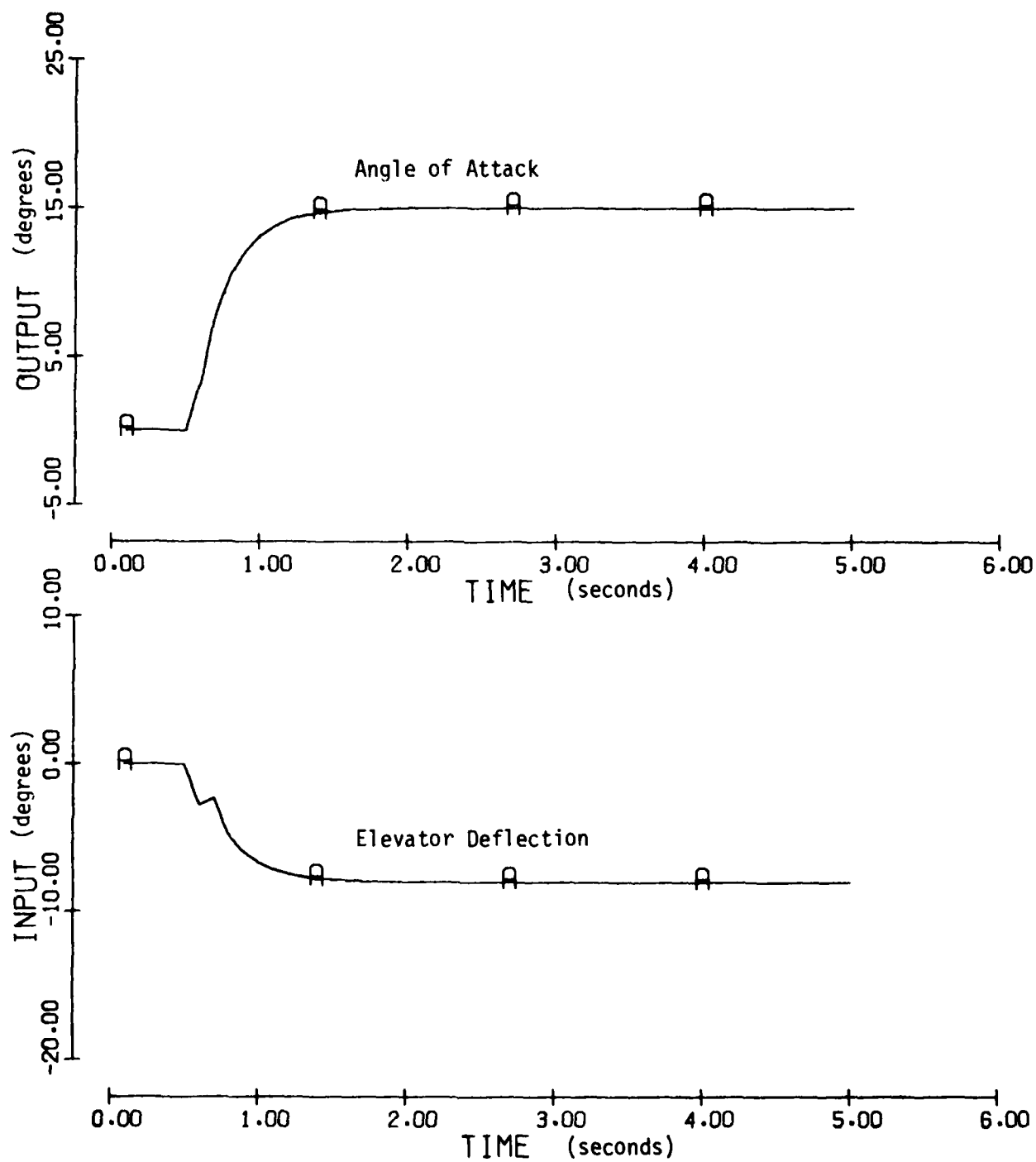
Figure 37a.   Gradient algorithm with two endpoints.

151
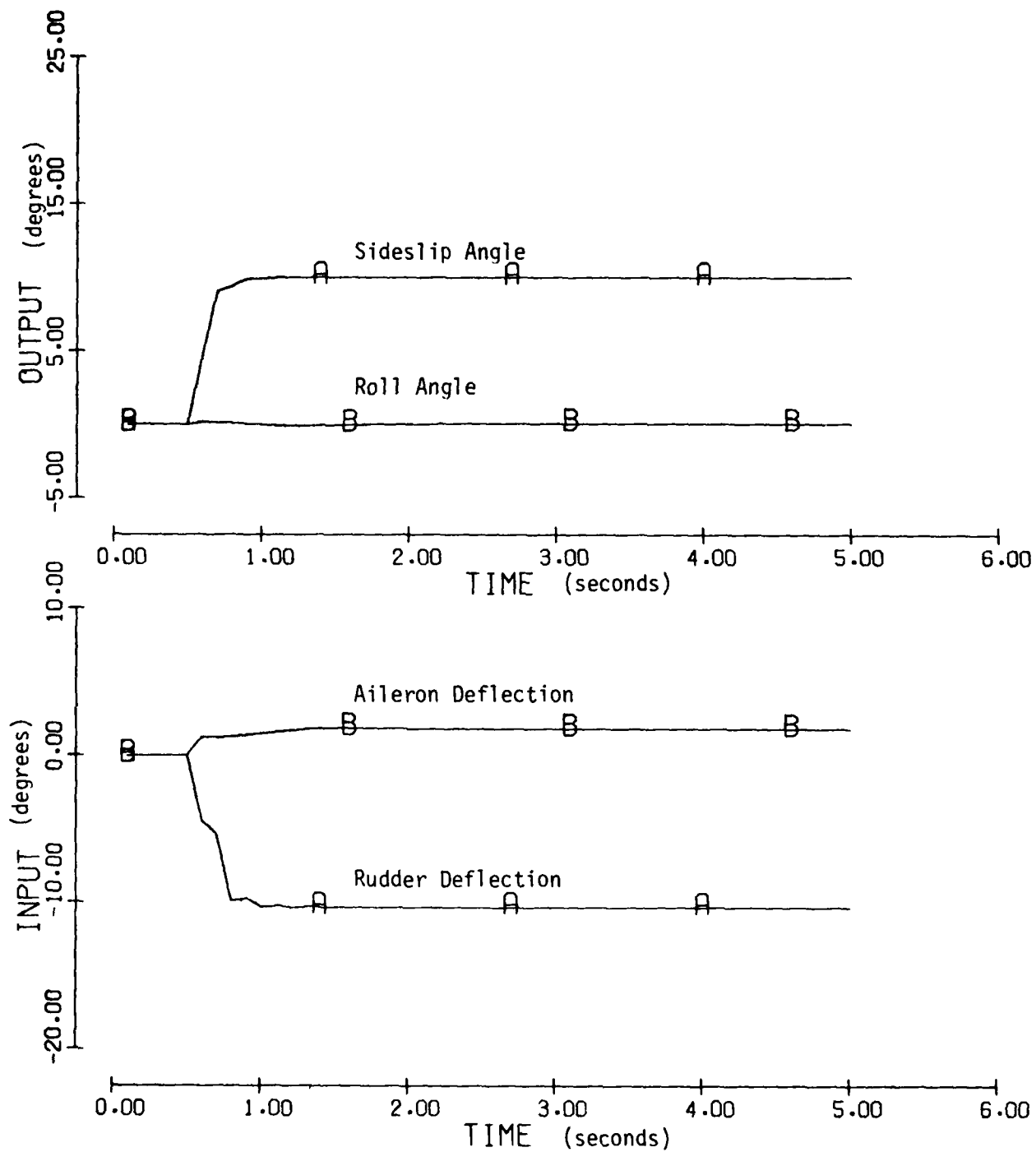
Figure 37b. Gradient algorithm with two endpoints.

152
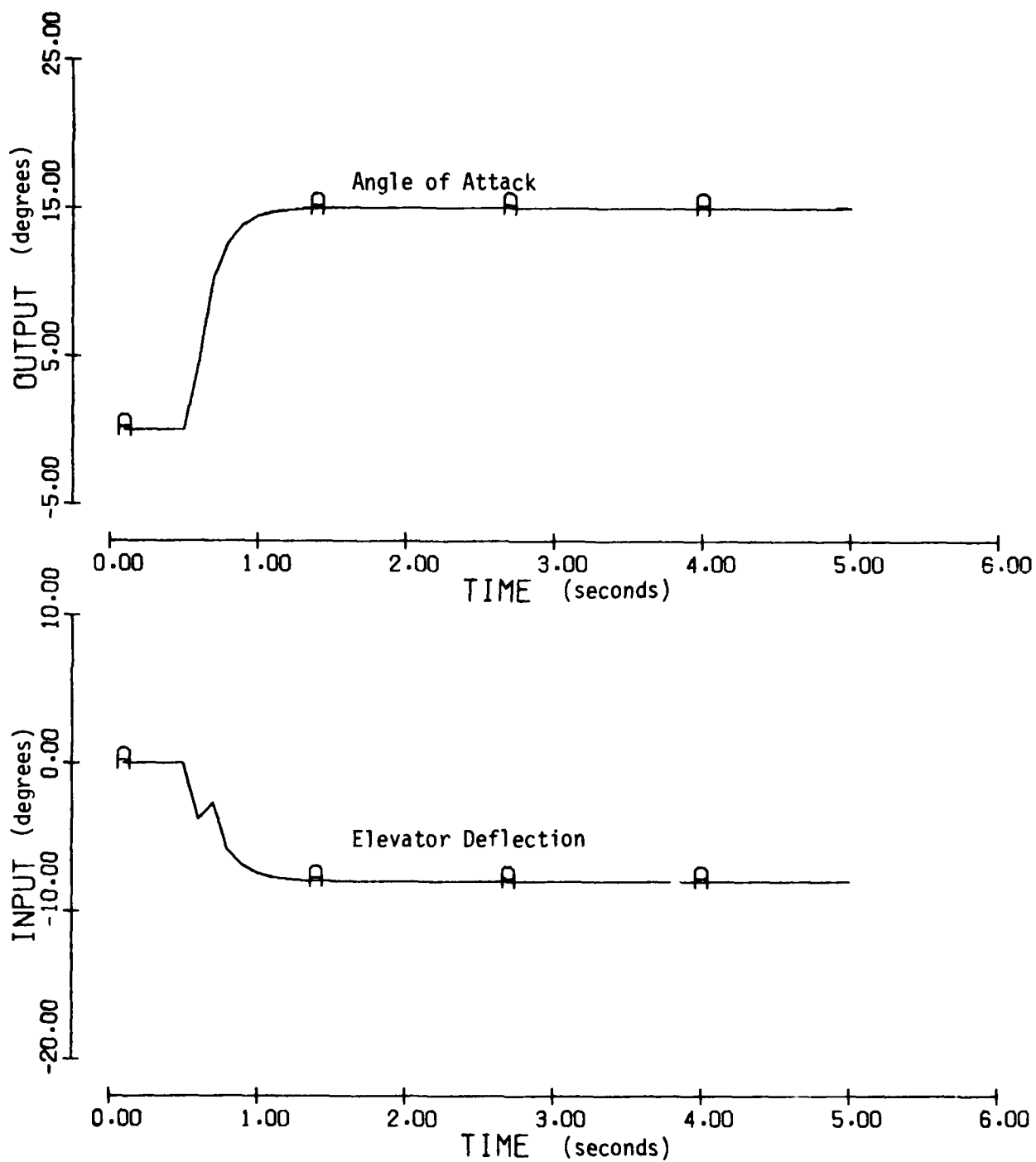
Figure 38a. Gradient algorithm with three endpoints, long horizon.

153
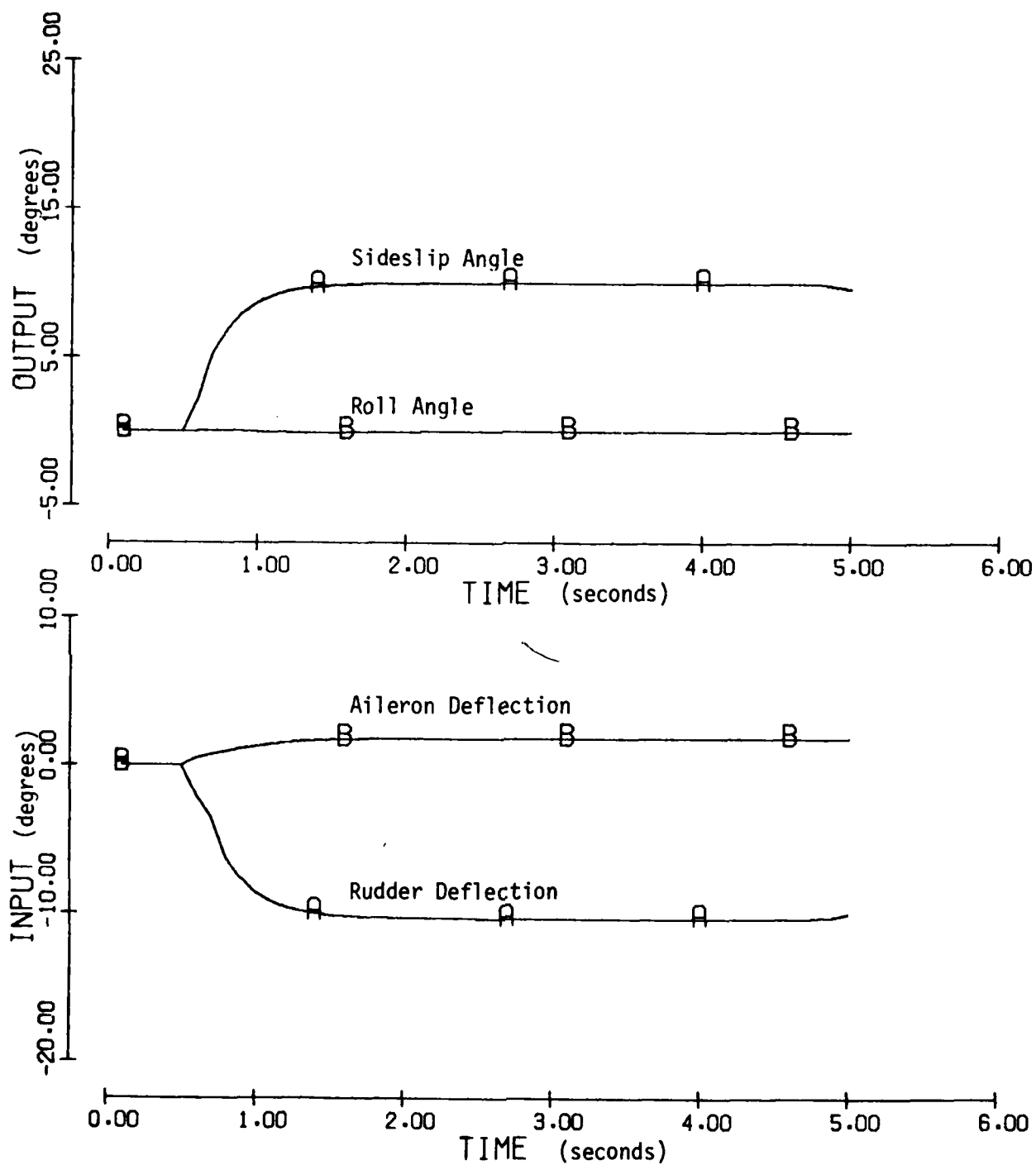
Figure 38b.   Gradient algorithm with three endpoints, long horizon.
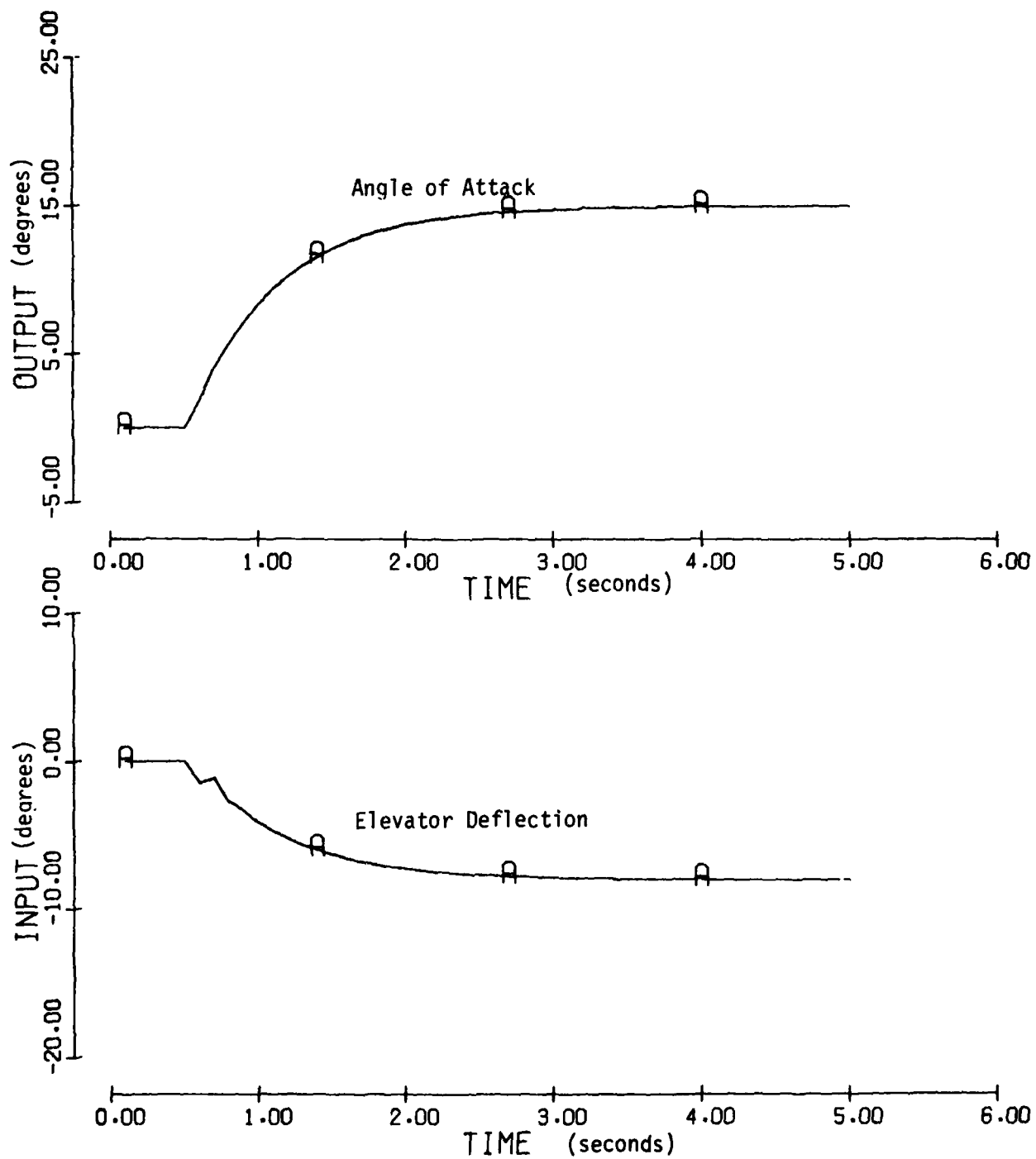
Figure 39a. Gradient algorithm, slow trajectory.

155
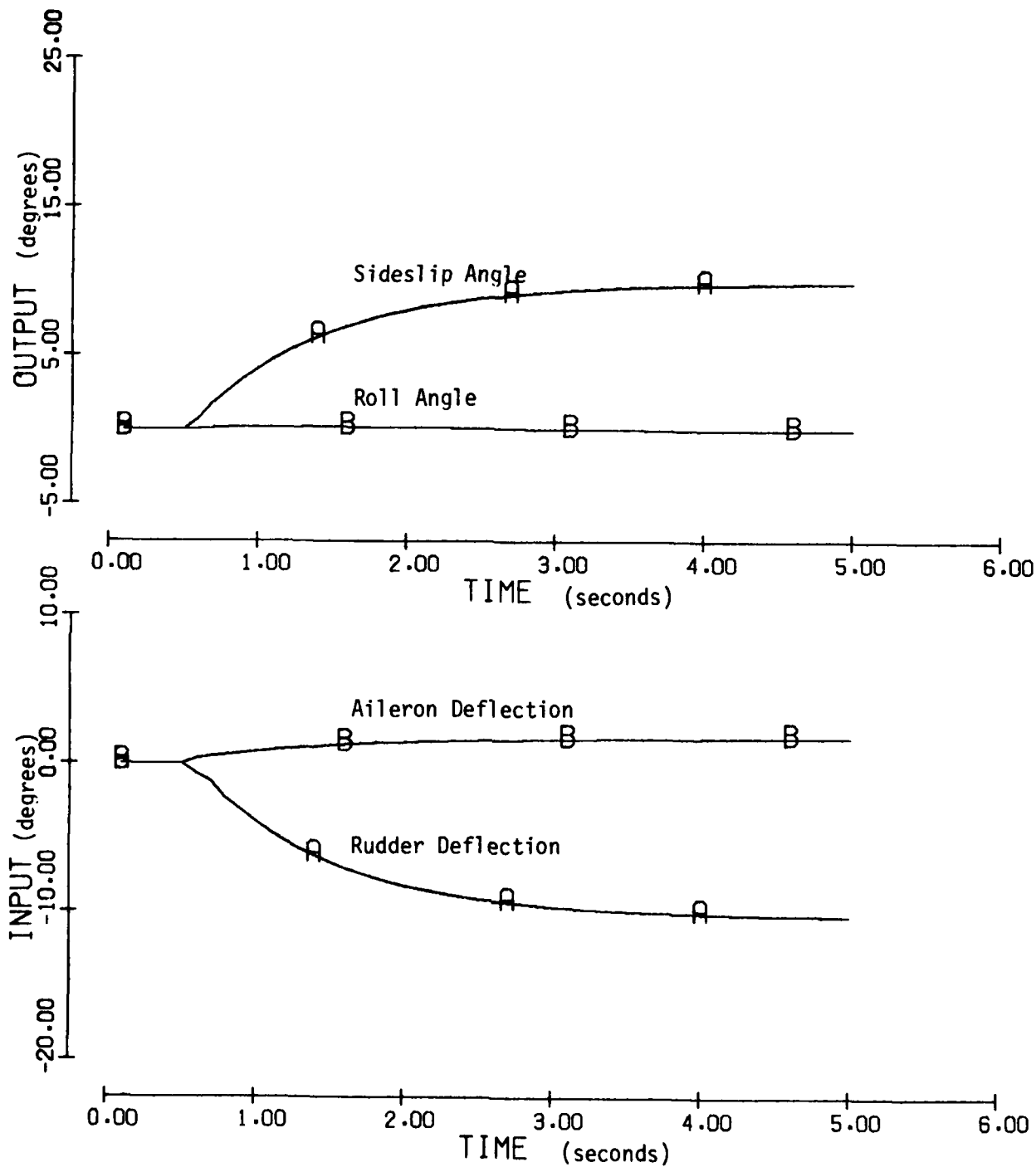
Figure 39b. Gradient algorithm, slow trajectory.
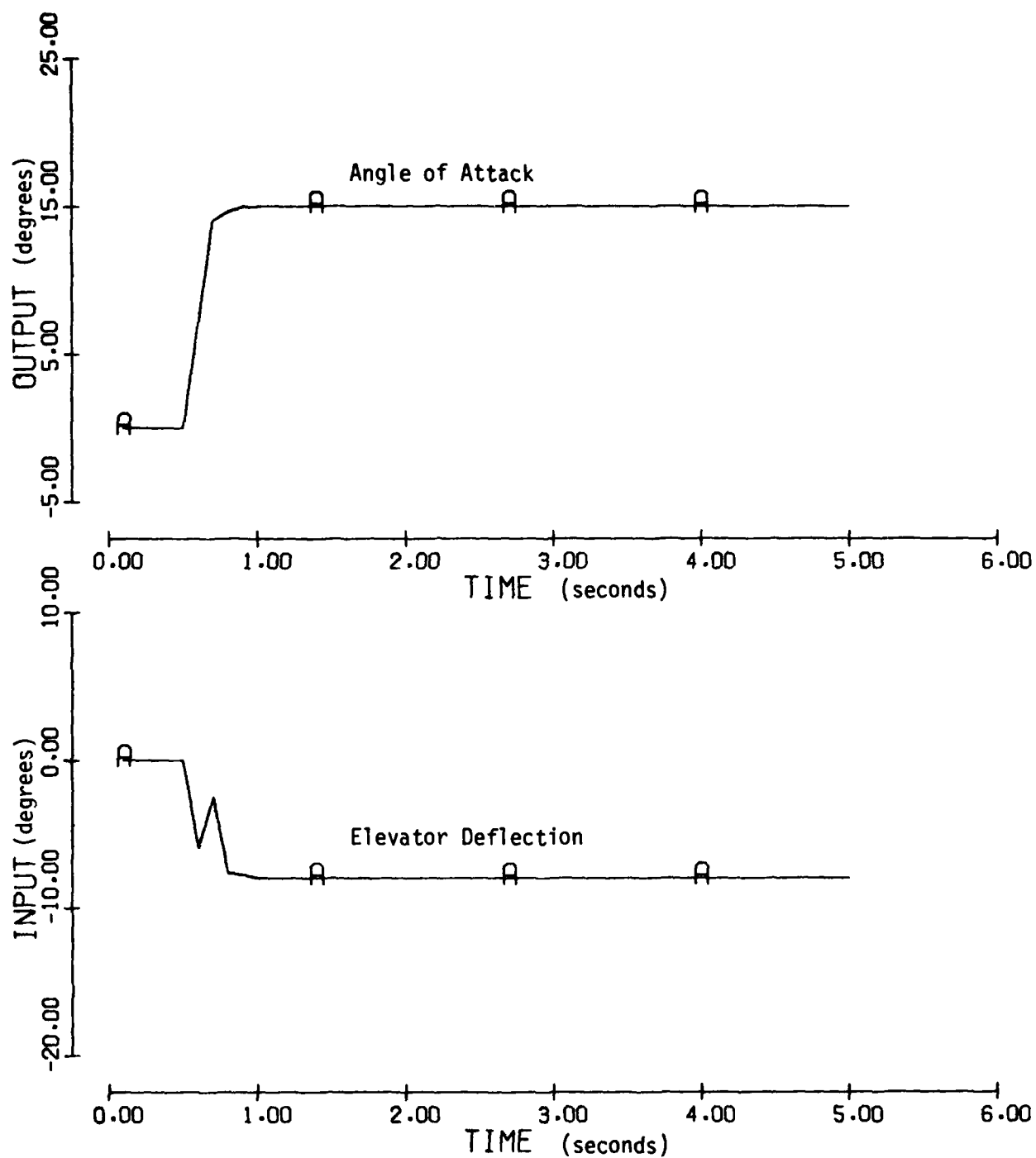
Figure 40a. Gradient algorithm, fast trajectory.
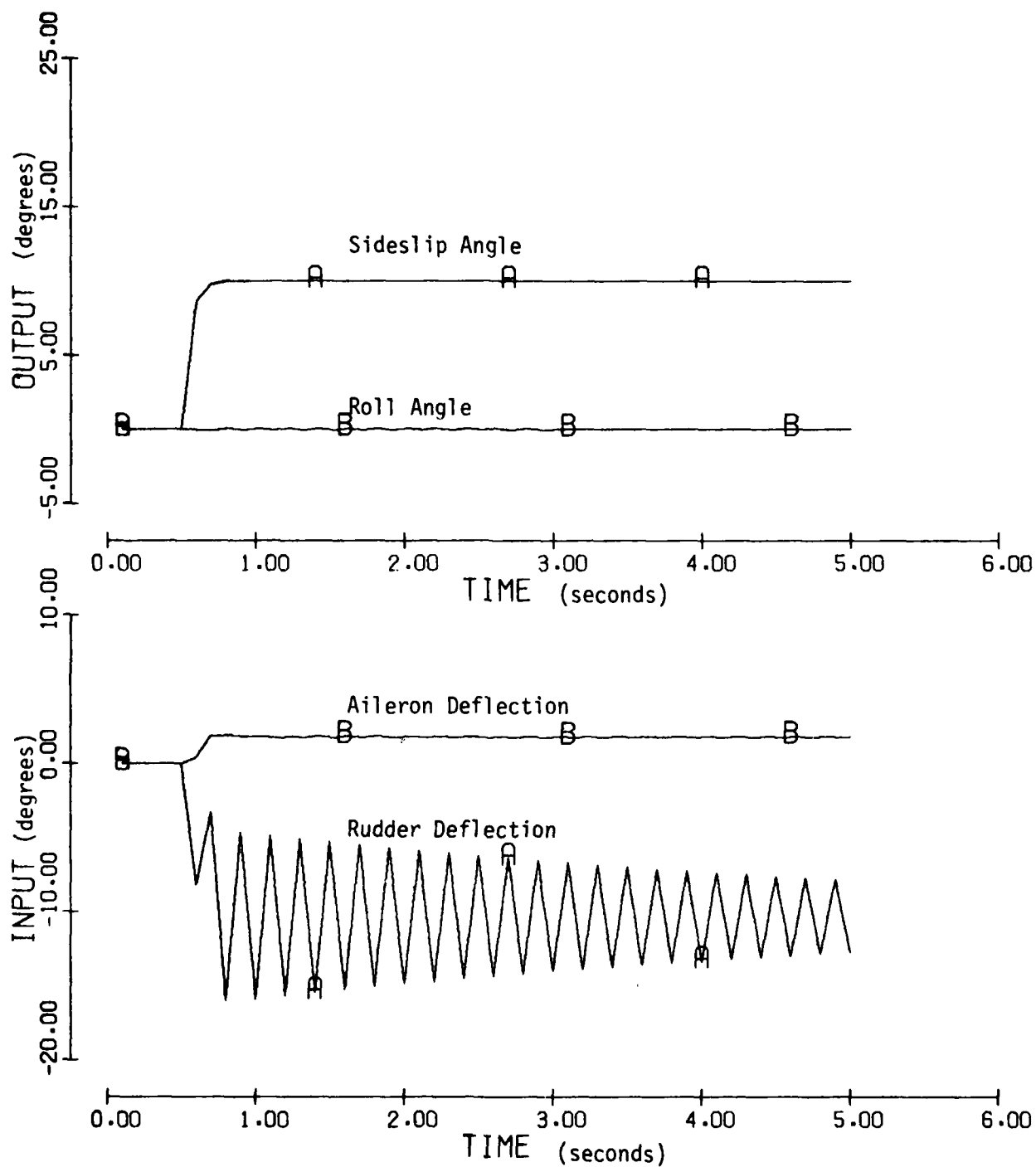
157
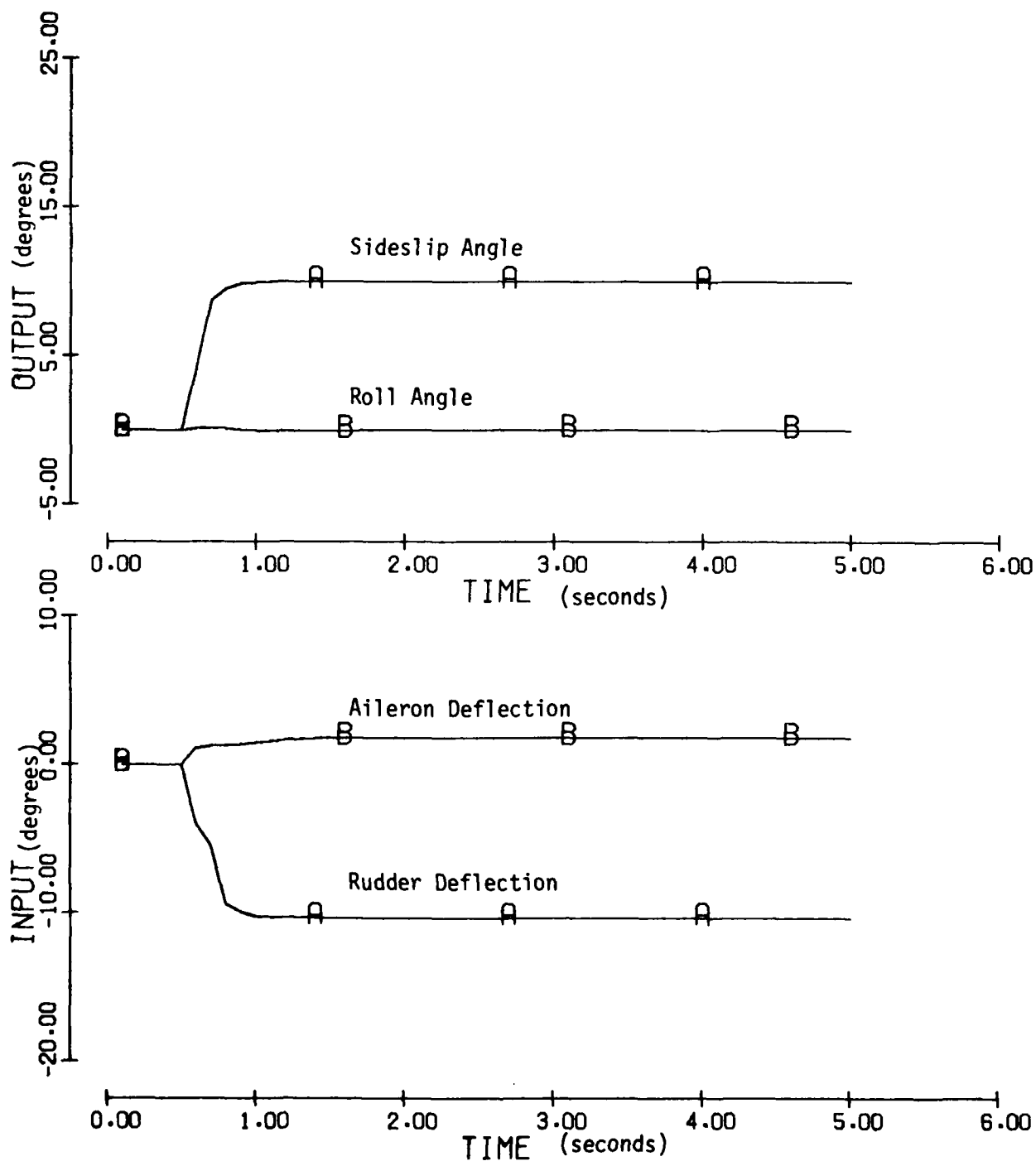
Figure 40b. Gradient algorithm, fast trajectory.
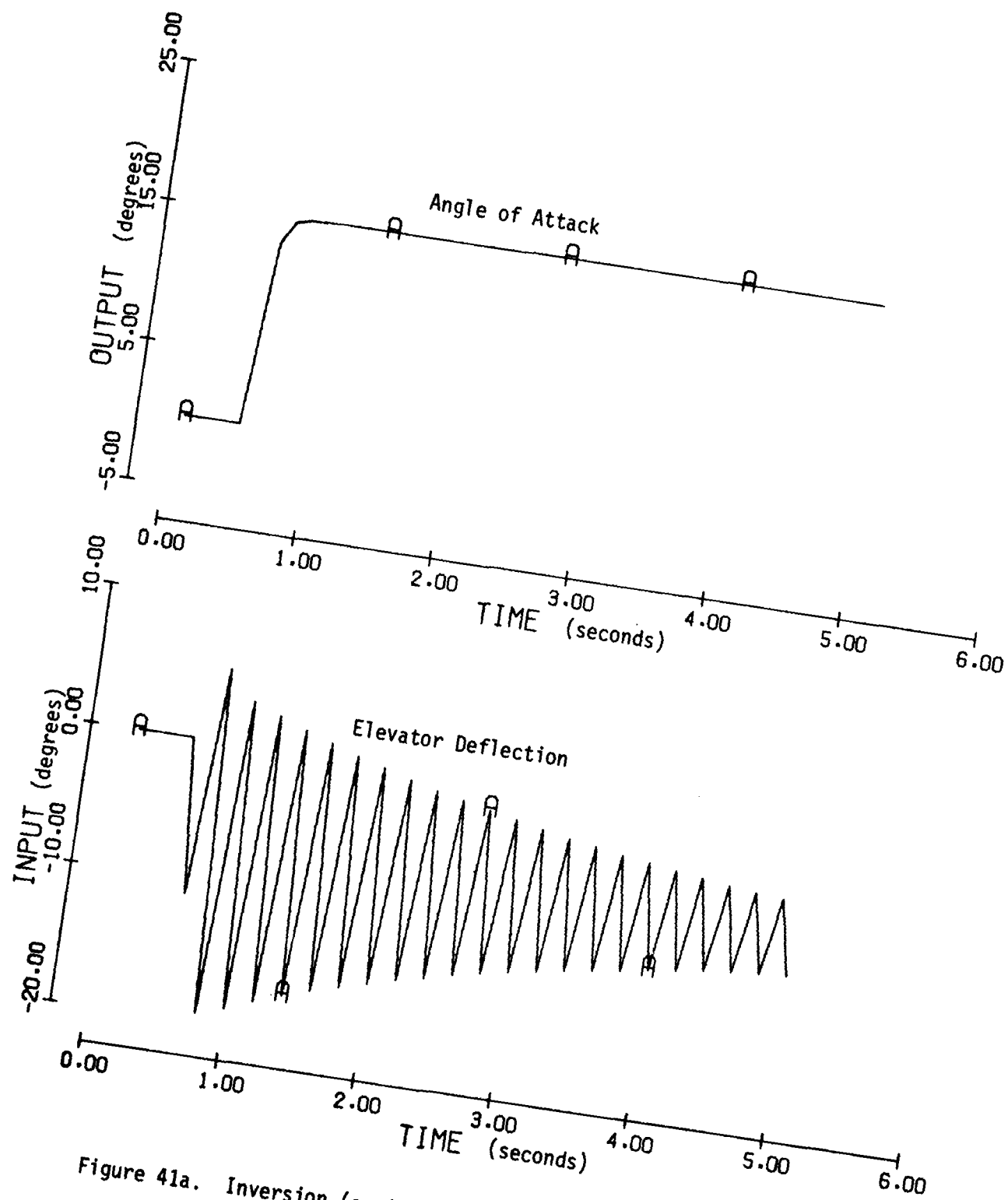
Figure 41a. Inversion (equivalent) algorithm, fast trajectory.

159

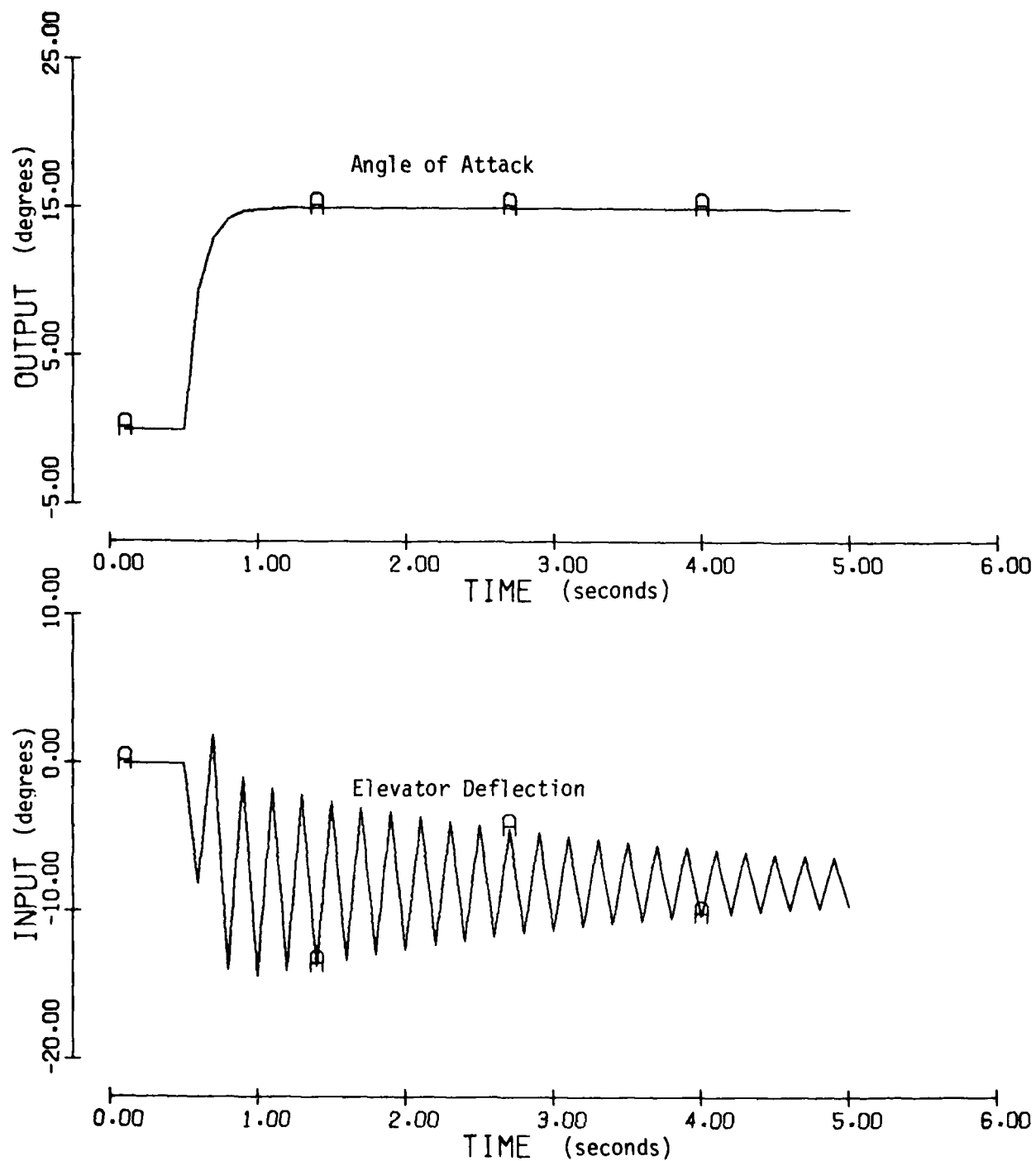Figure 41b. Inversion (equivalent) algorithm, fast trajectory.

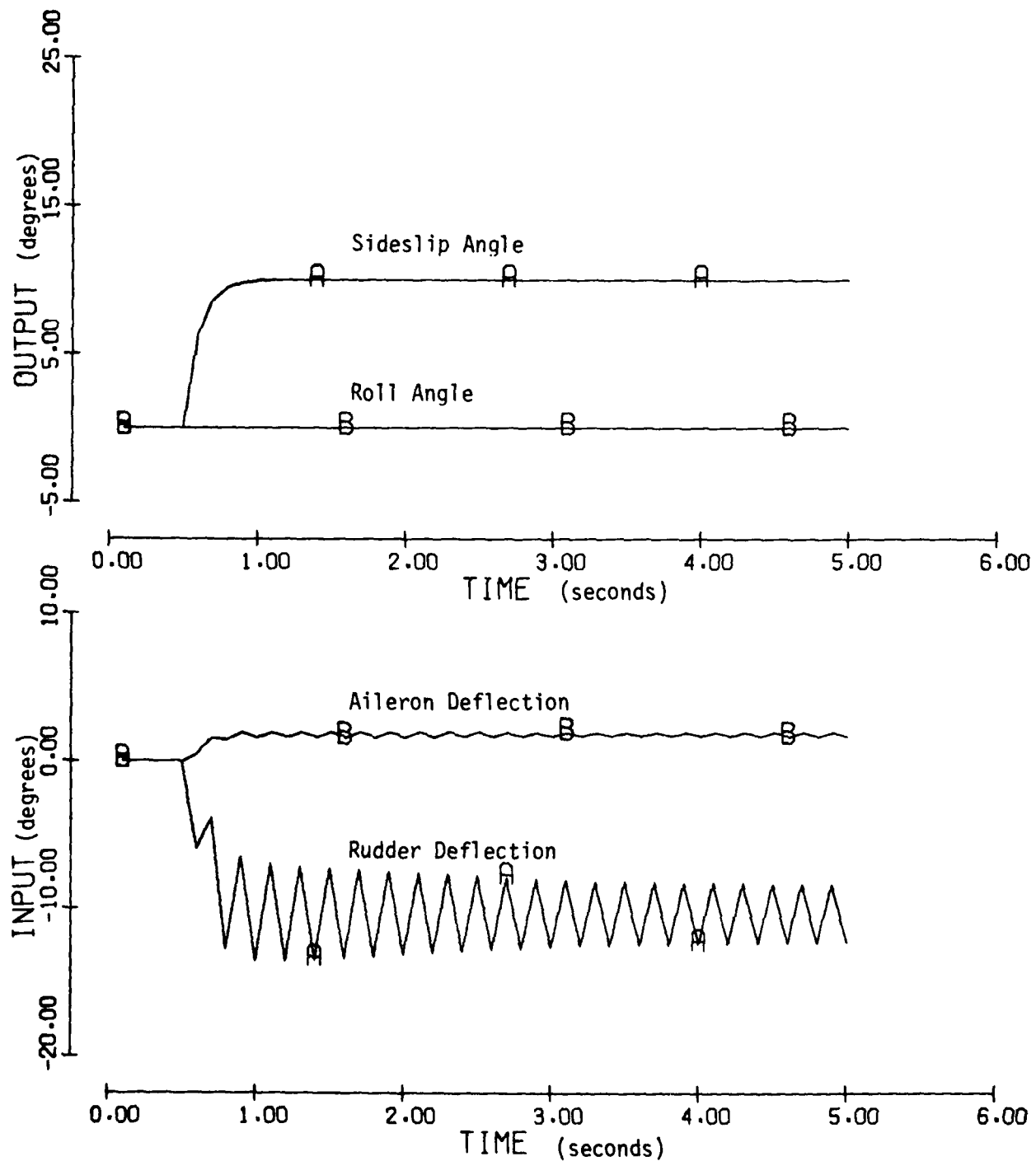Figure 42a. Inversion algorithm, 250°/second rate limit.

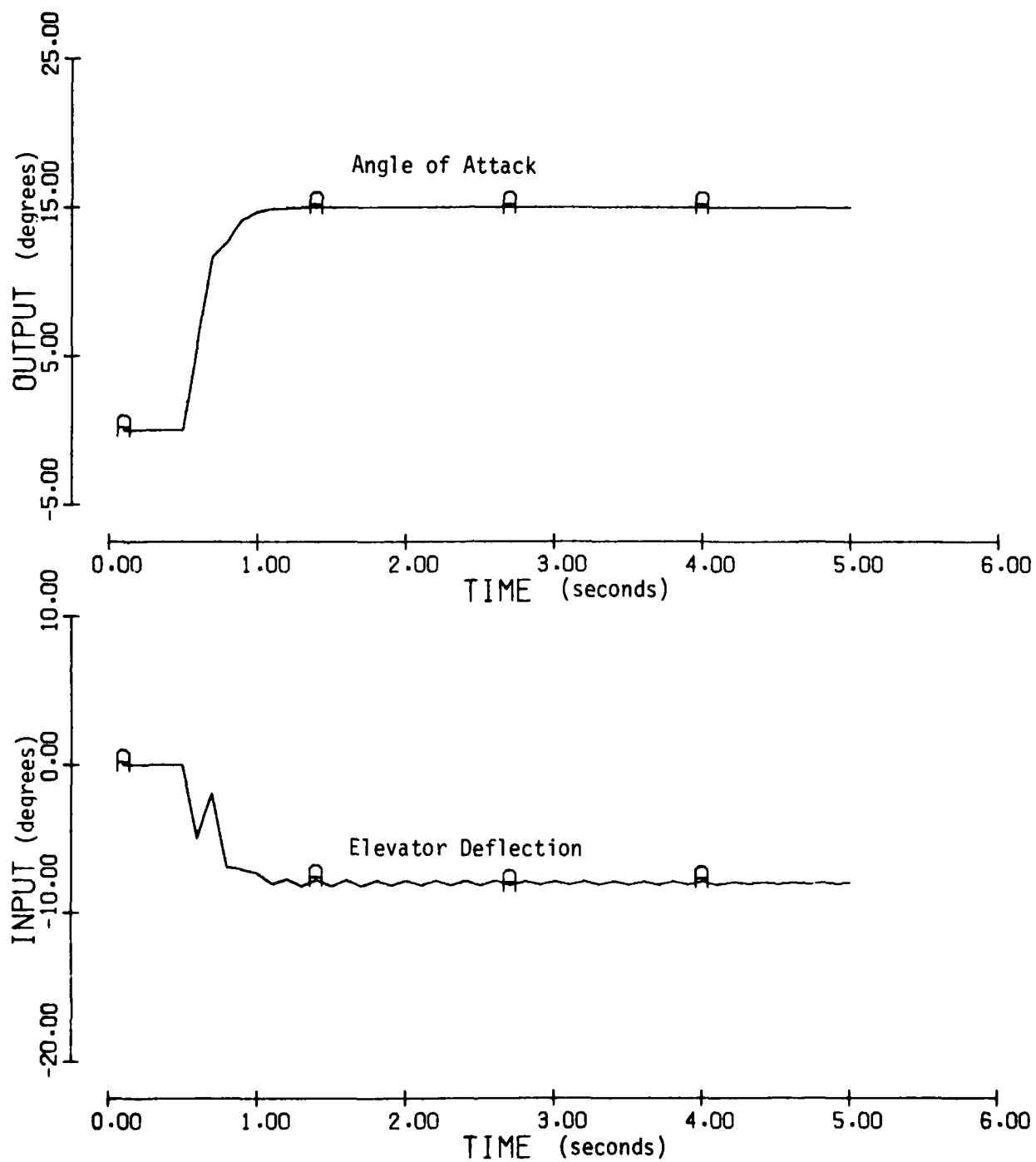Figure 42b.   Inversion algorithm, 250°/second rate limit.

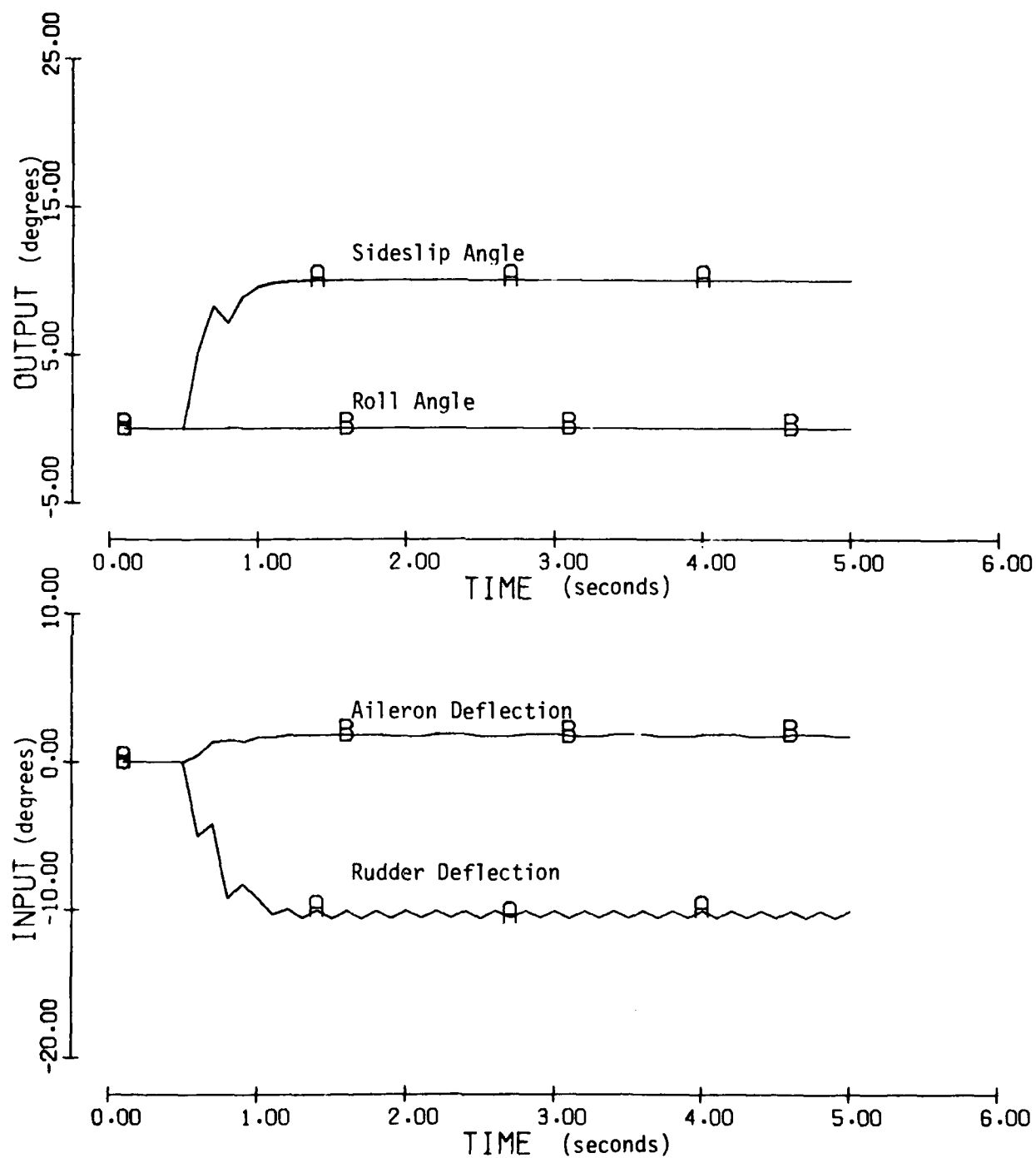Figure 43a. Inversion algorithm, 50°/second rate limit.

163

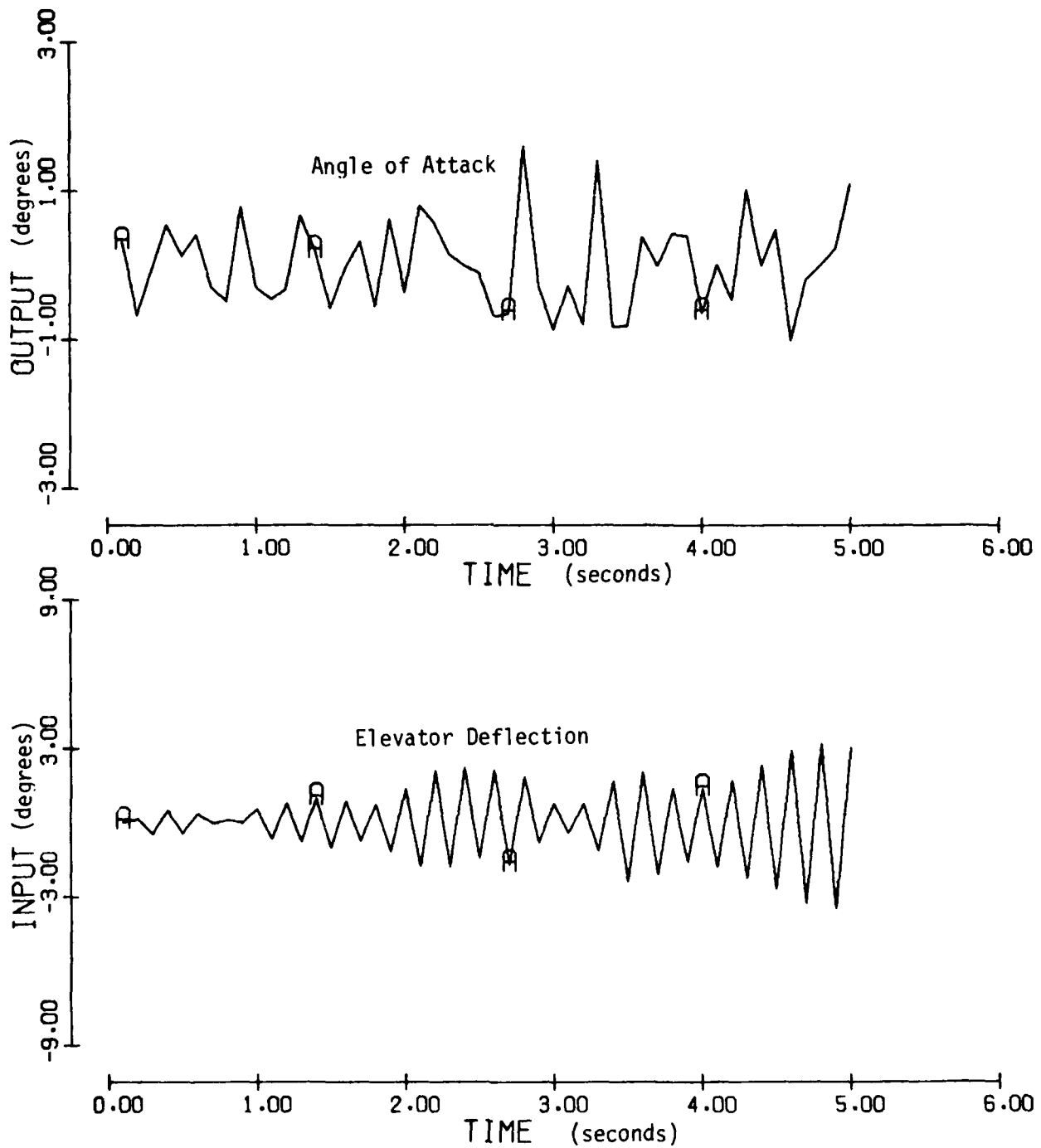Figure 43b. Inversion algorithm, 50°/second rate limit.

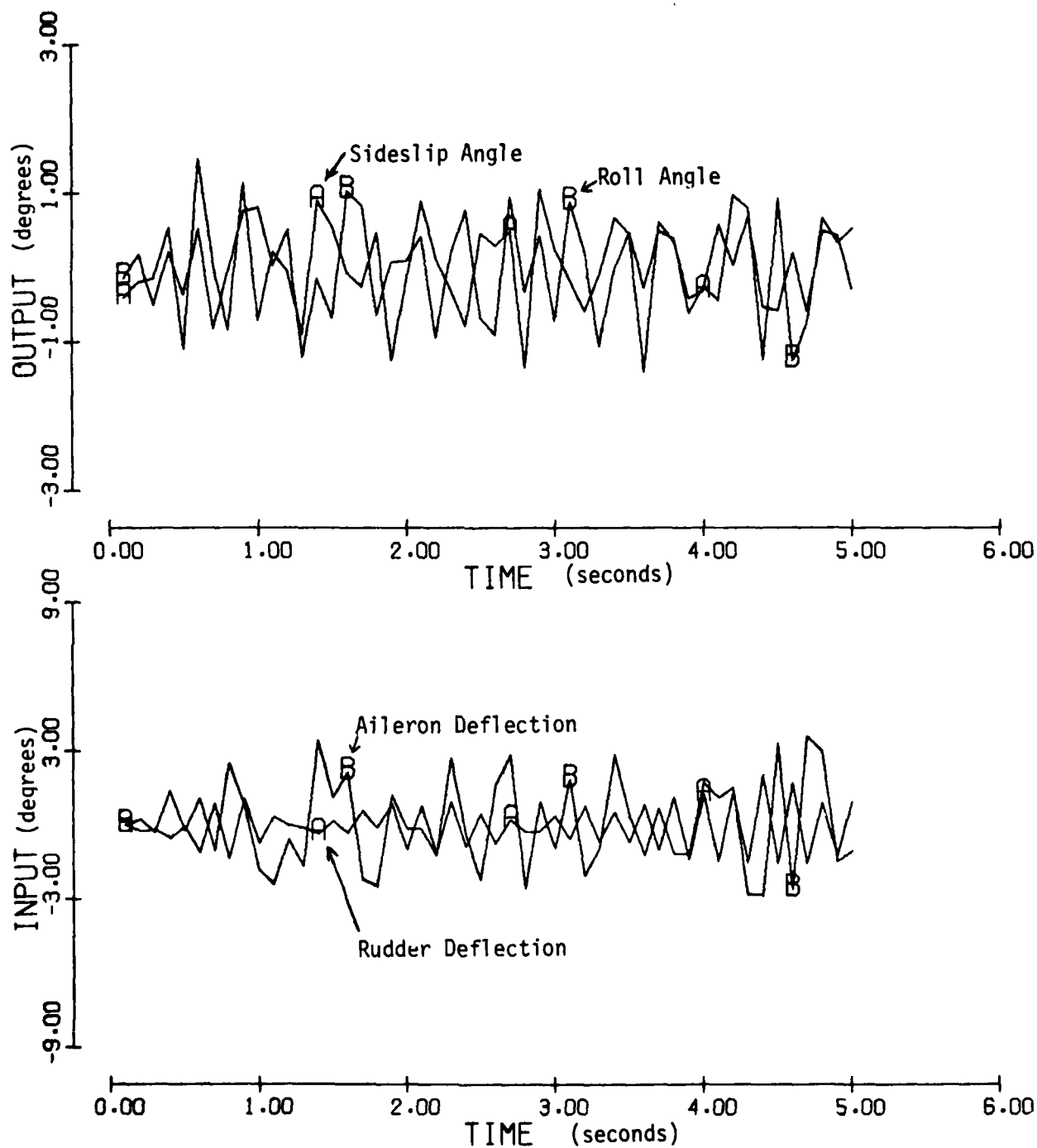Figure 44a.   Inversion algorithm with measurement noise.

Figure 44b.   Inversion algorithm with measurement noise.
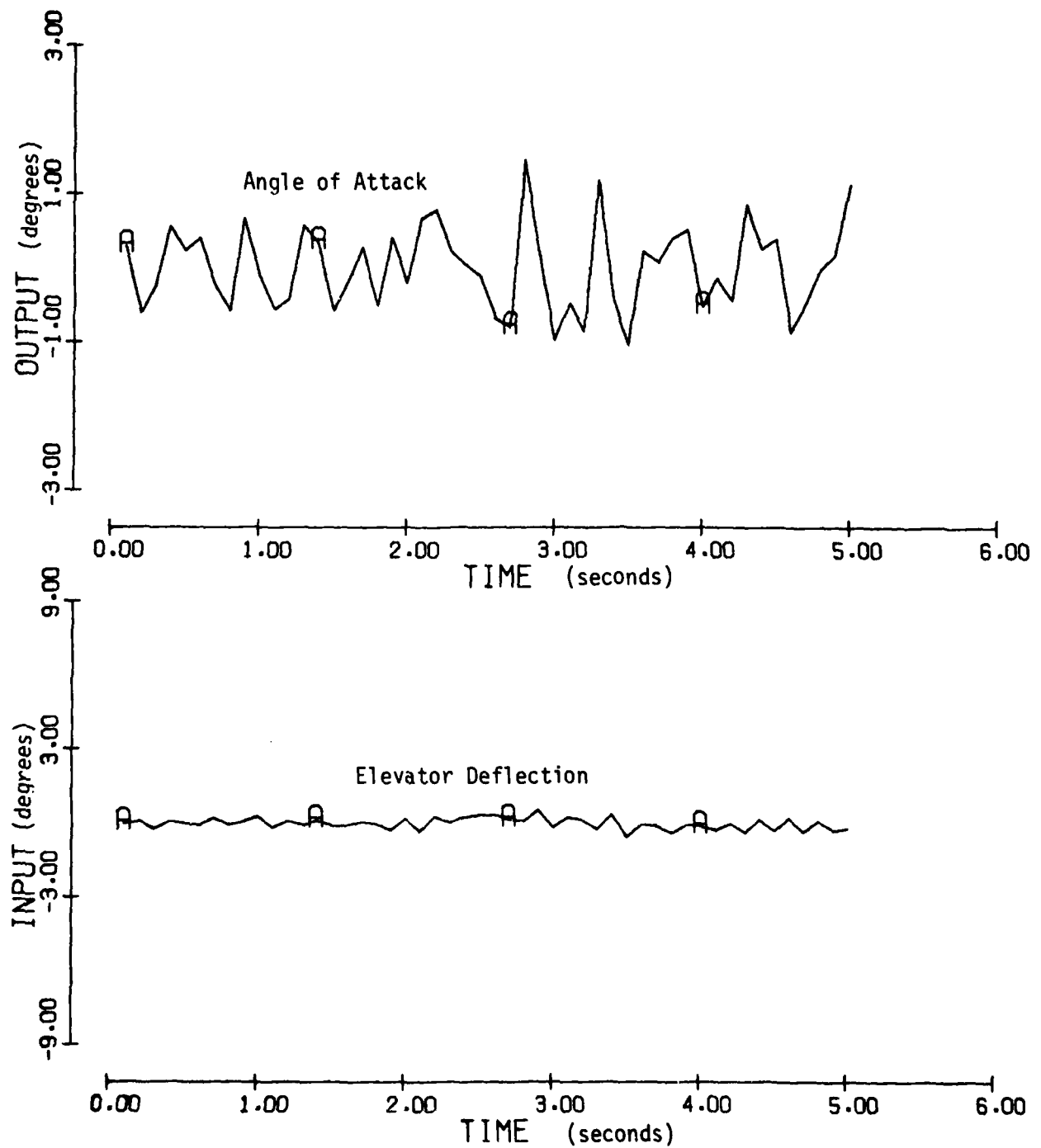
Figure 45a. Gradient algorithm with measurement noise.

167
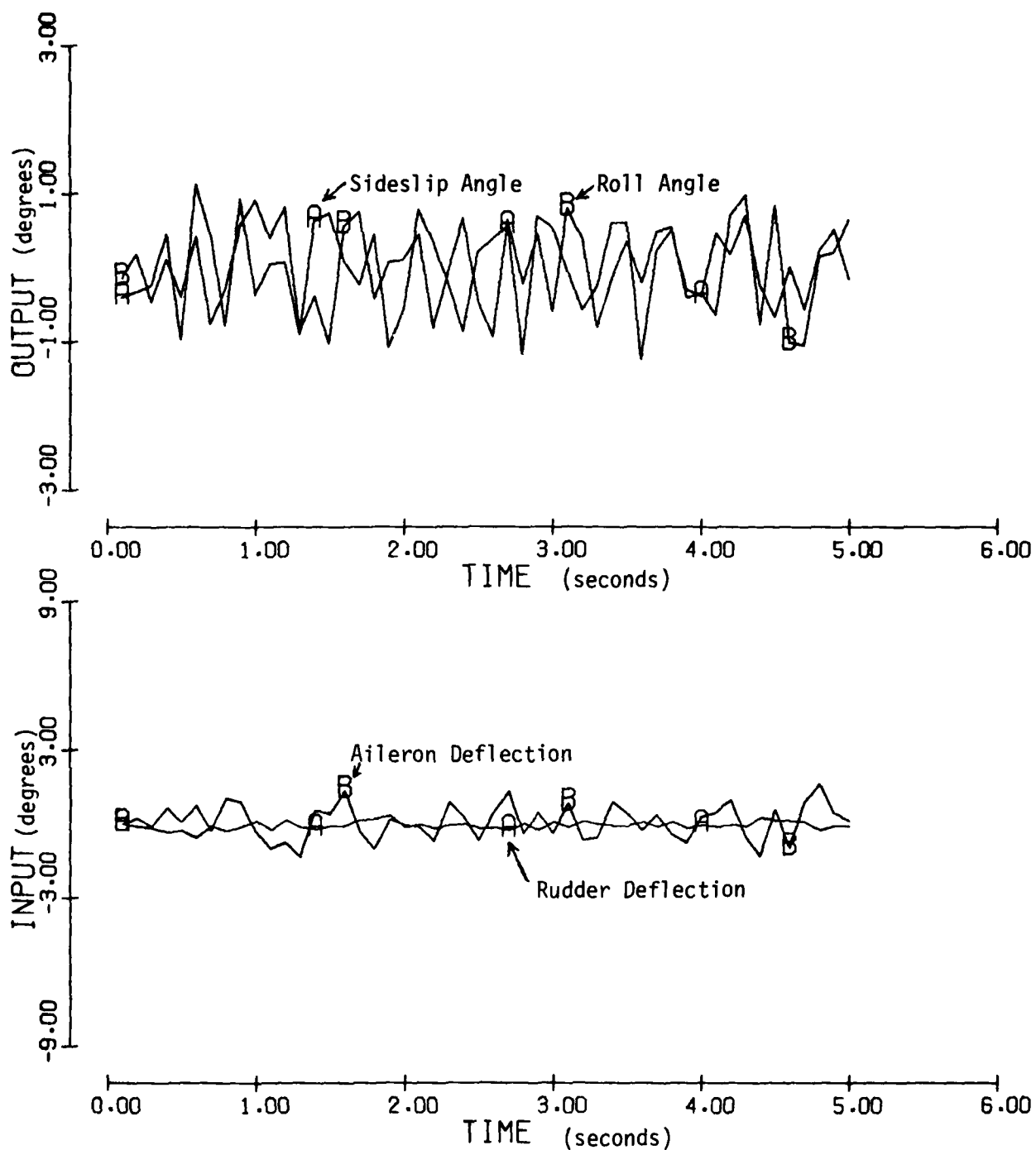
Figure 45b. Gradient algorithm with measurement noise.

168

Figure 46a.   Inversion algorithm with gain mismatch of 3, $\tau = .1$.

169
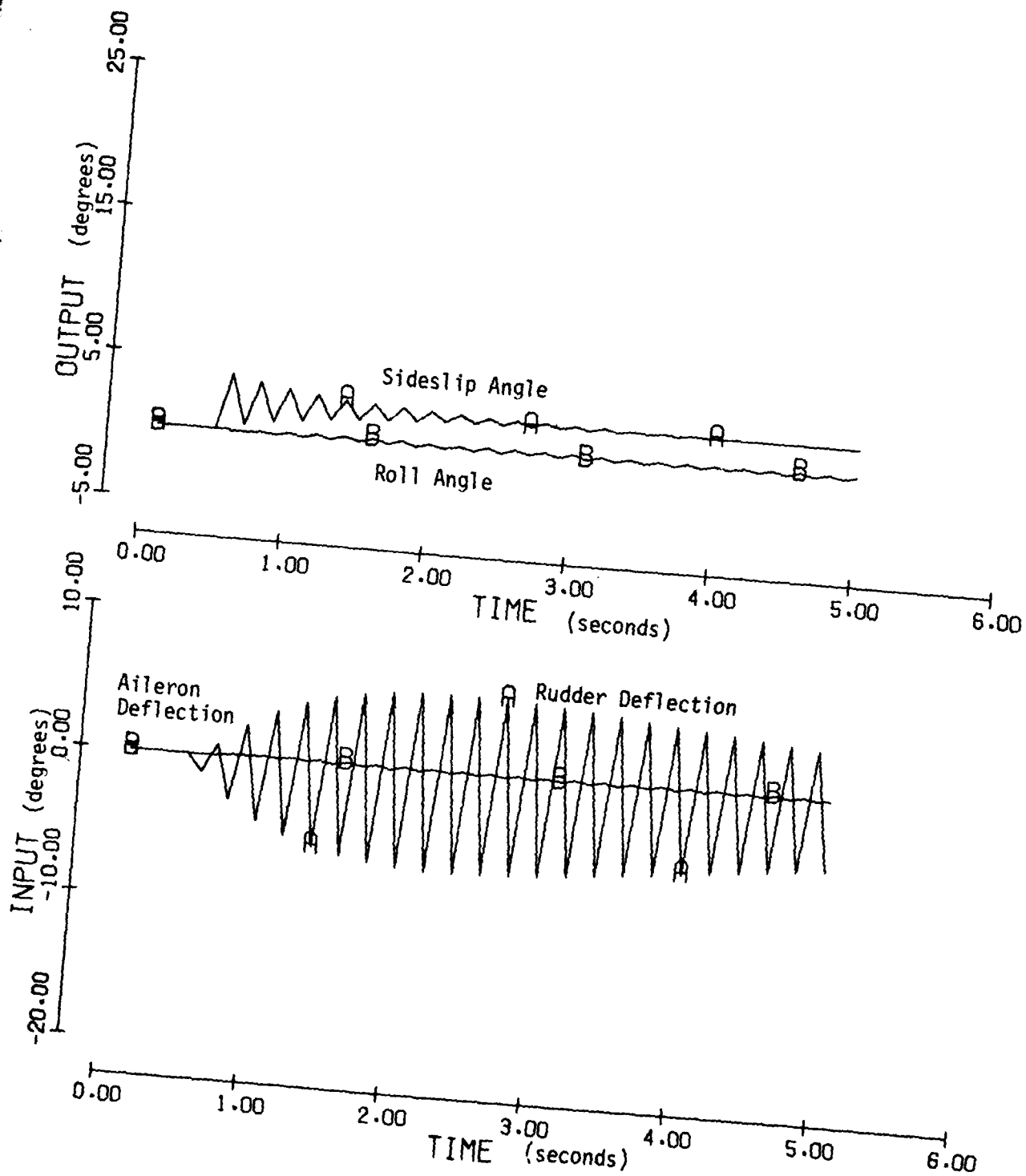
Figure 46b. Inversion algorithm with gain mismatch of 3, $\tau = .1$.

170
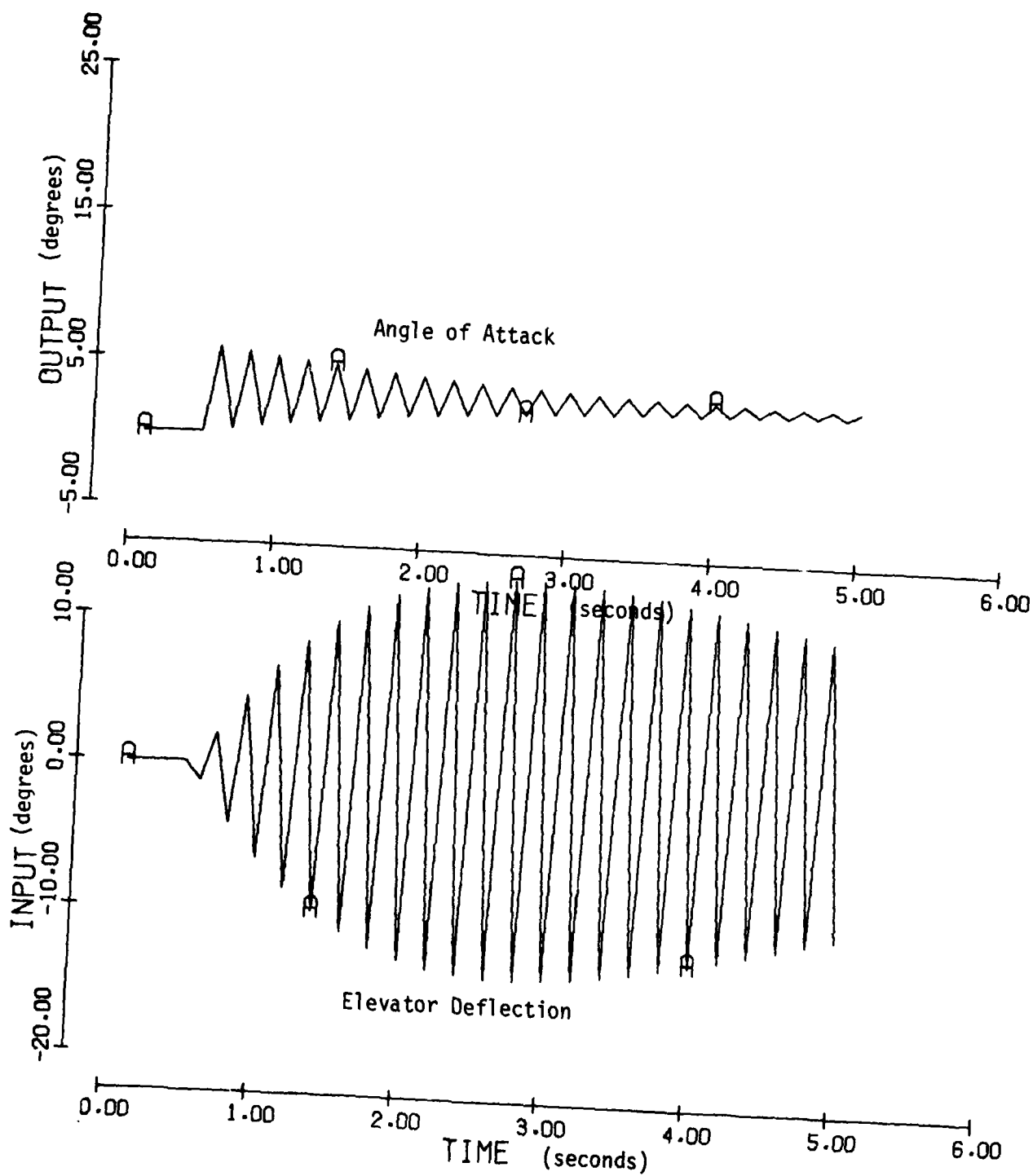
Figure 47a. Inversion algorithm with gain mismatch of 4, $\tau = .15$.

171
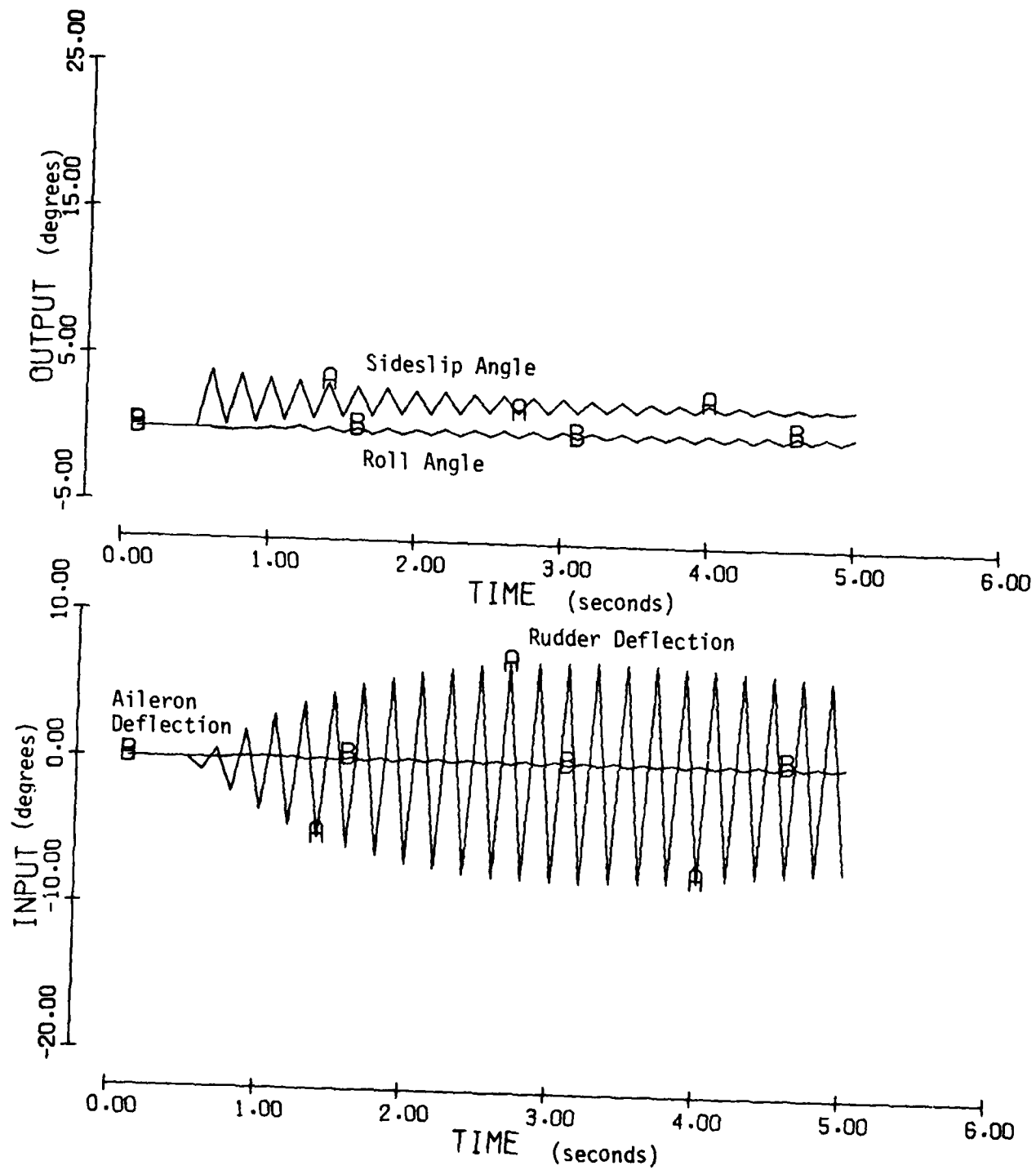
Figure 47b.   Inversion algorithm with gain mismatch of 4, $\tau = .15$.

172
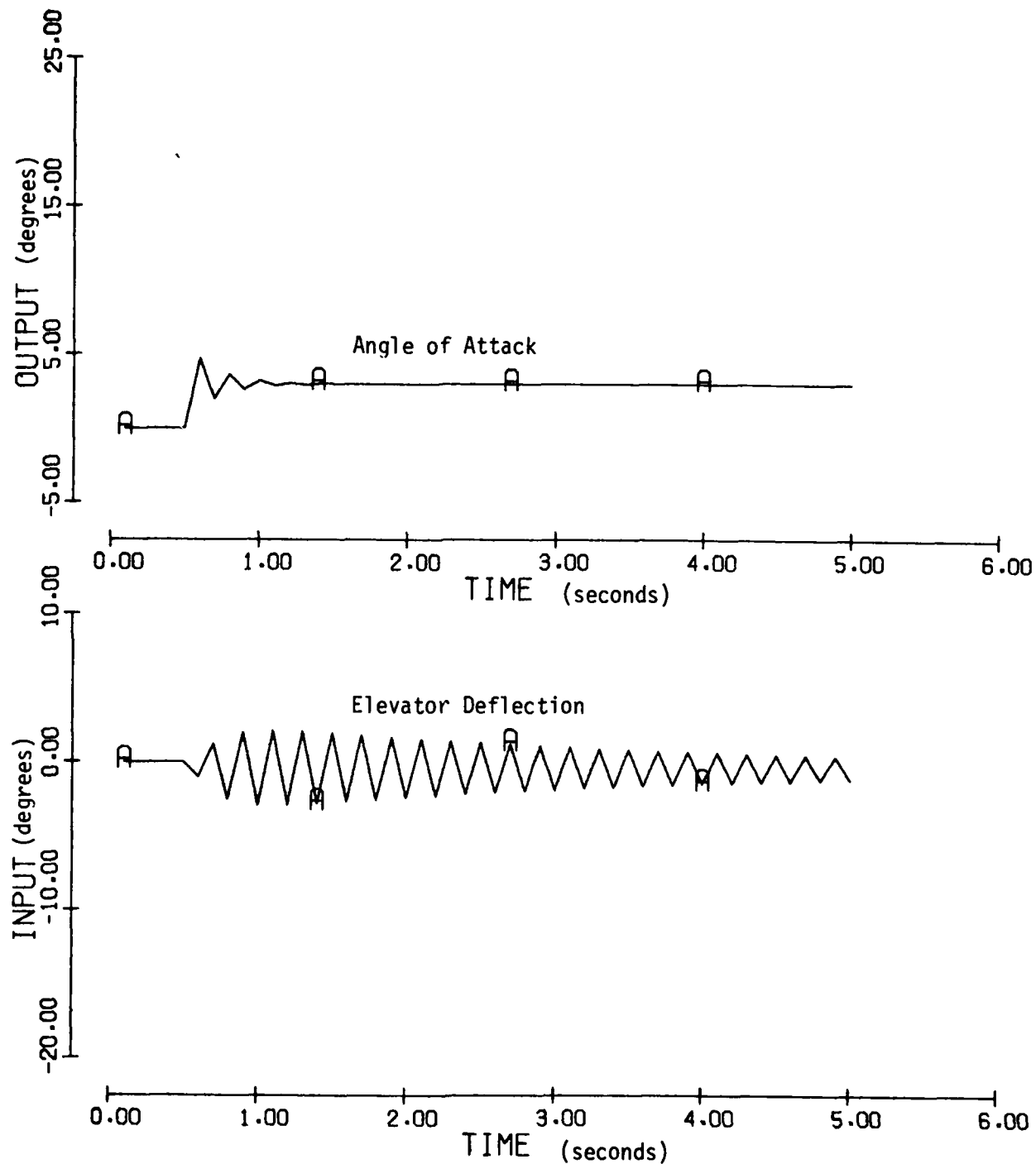
Figure 48a. Inversion algorithm with gain mismatch of 4, $\tau = .2$.

173
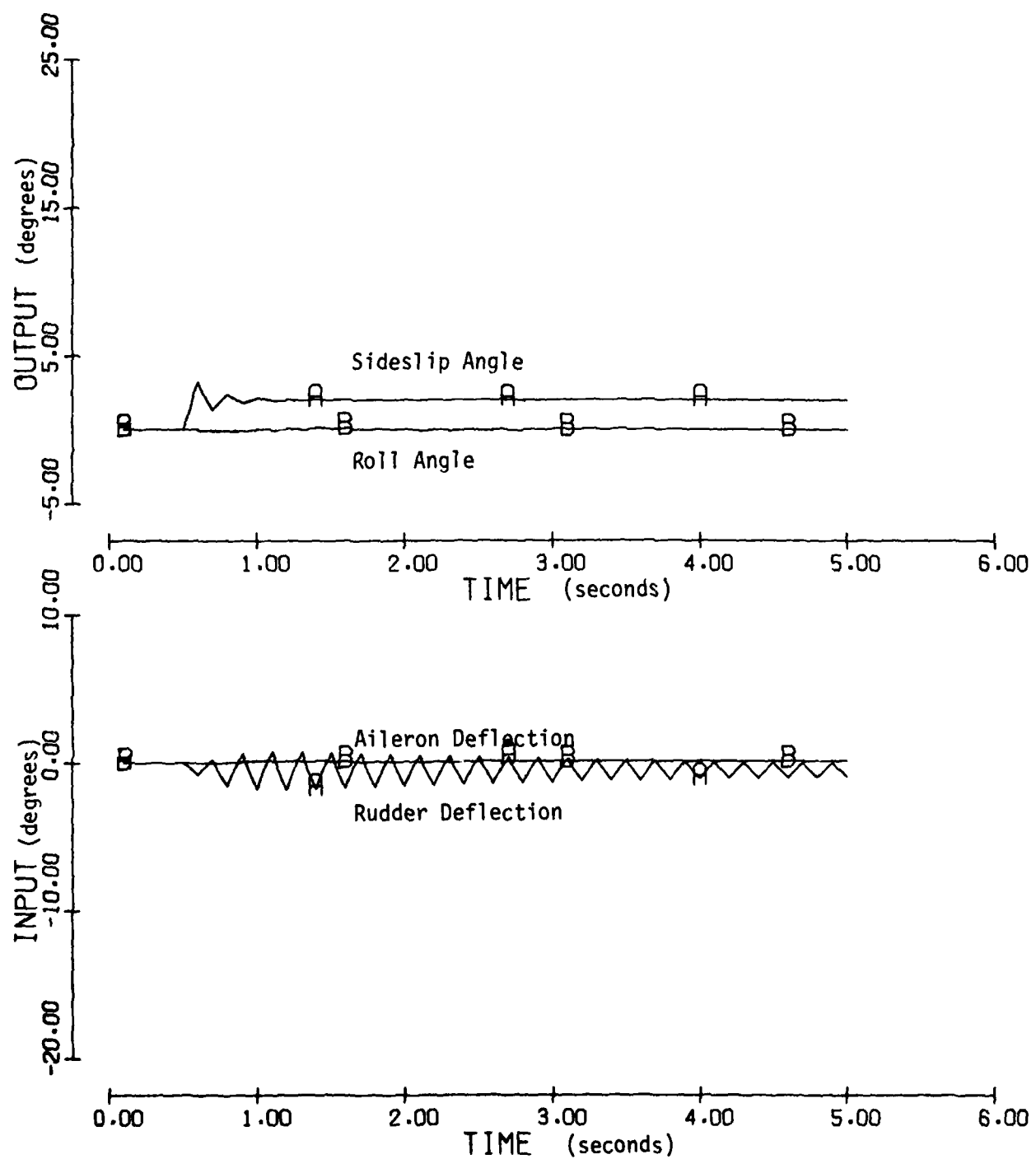
Figure 48b. Inversion algorithm with gain mismatch of 4, $\tau = .2$.

174
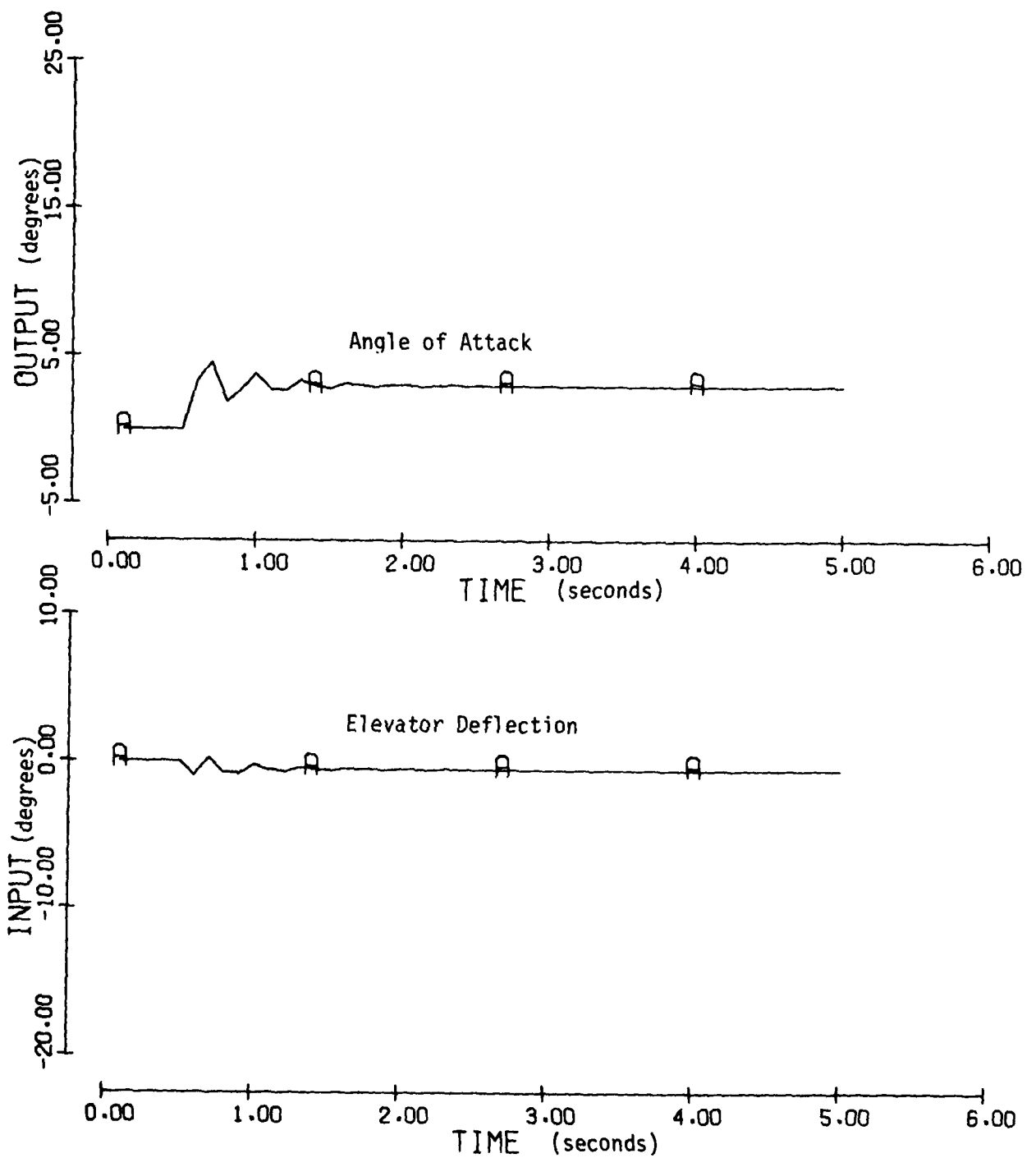
Figure 49a.   Gradient algorithm with gain mismatch of 3, τ = .1.
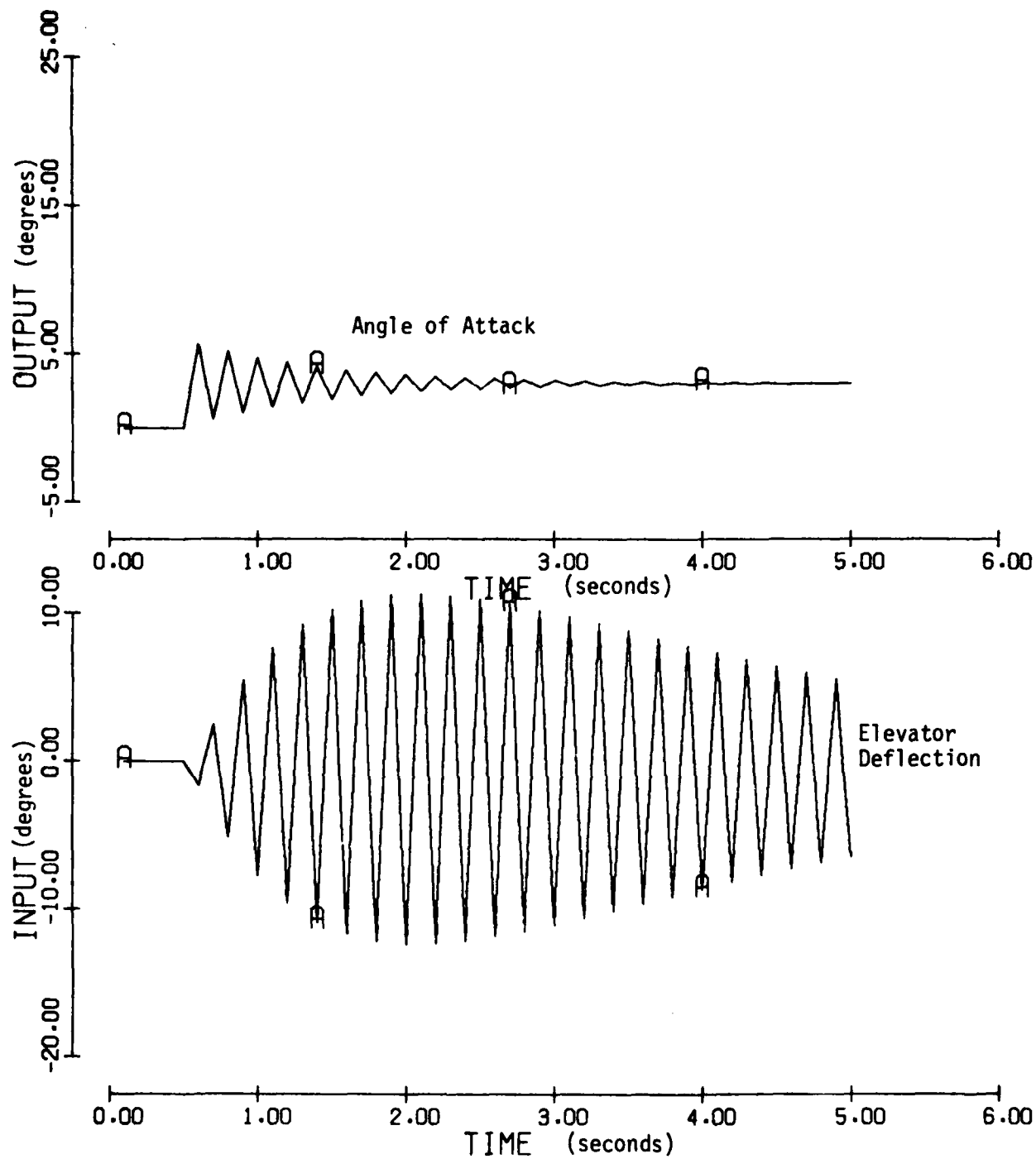
175
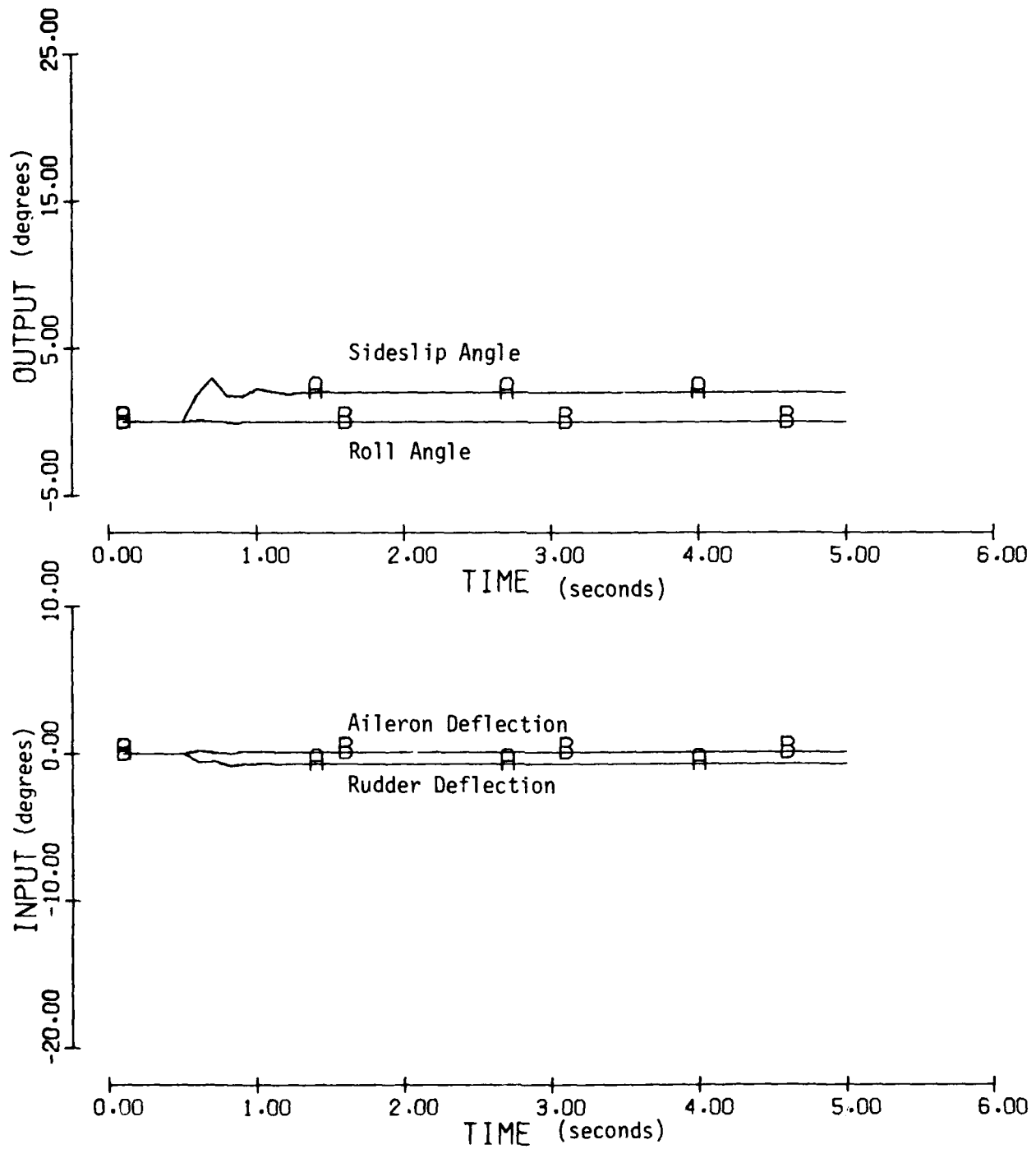
Figure 49b. Gradient algorithm with gain mismatch of 3, $\tau = .1$.
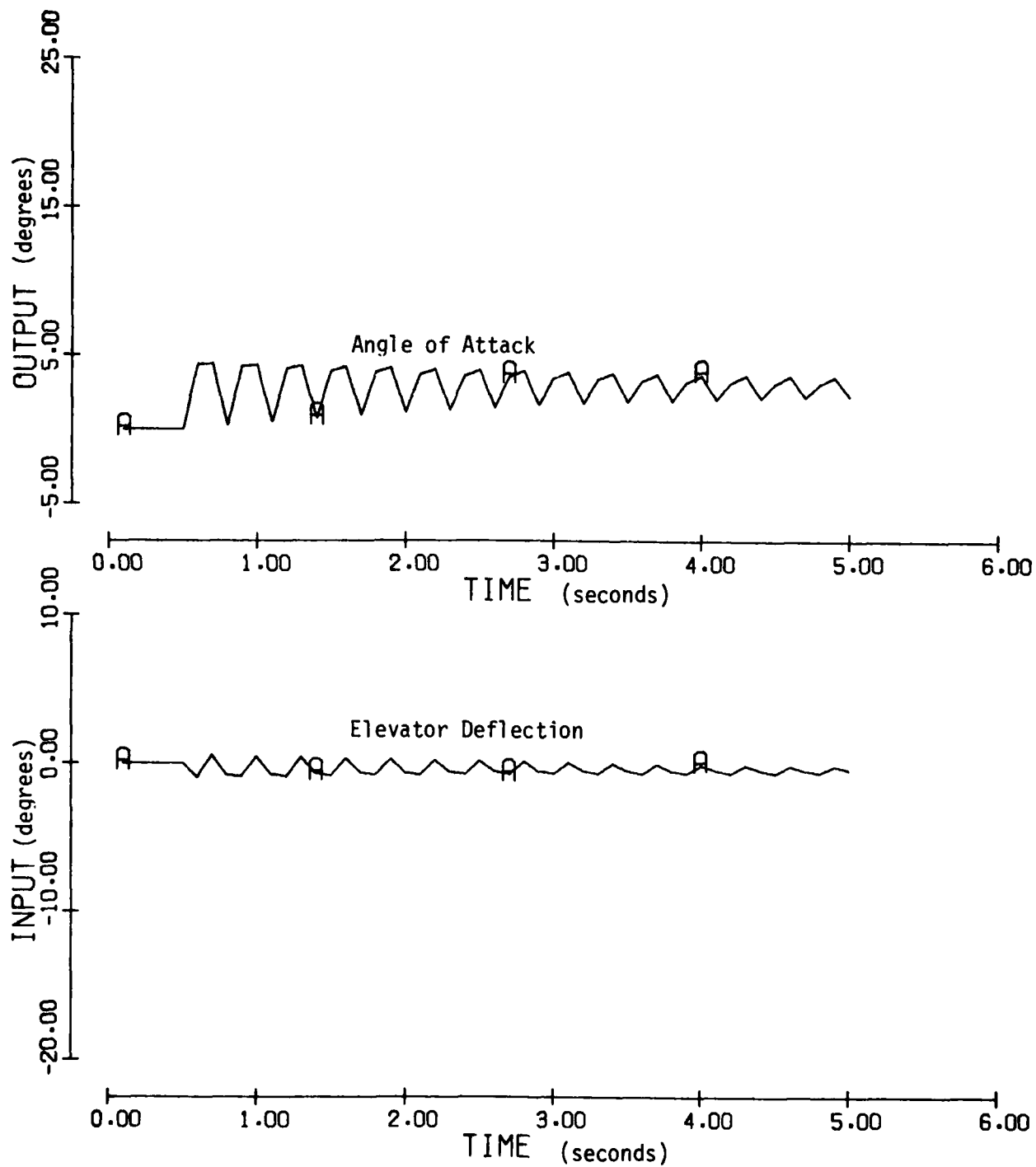
Figure 50a.   Gradient algorithm with gain mismatch of 4, τ = .1.
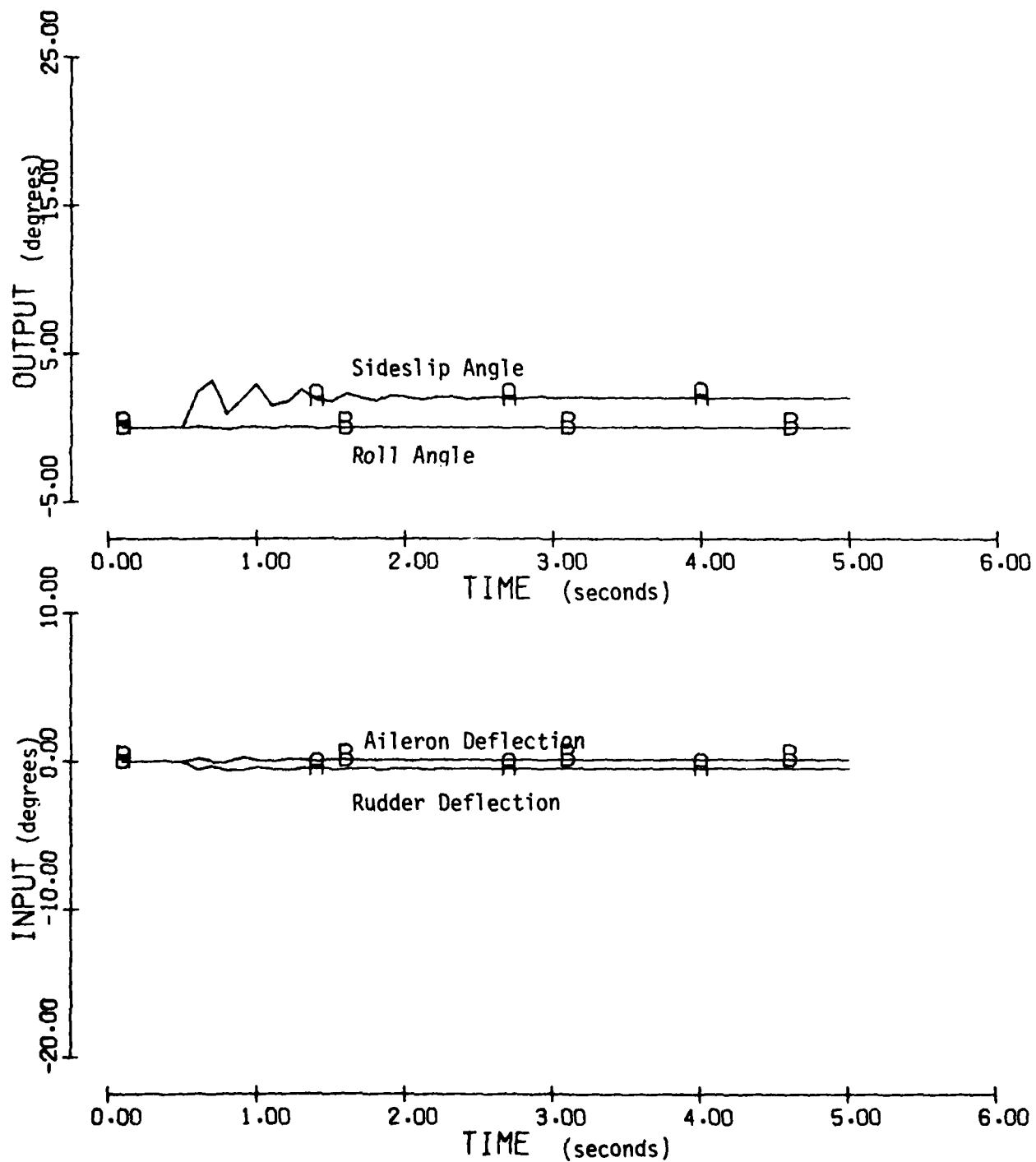
Figure 50b.   Gradient algorithm with gain mismatch of 4, $\tau = .1$.

178
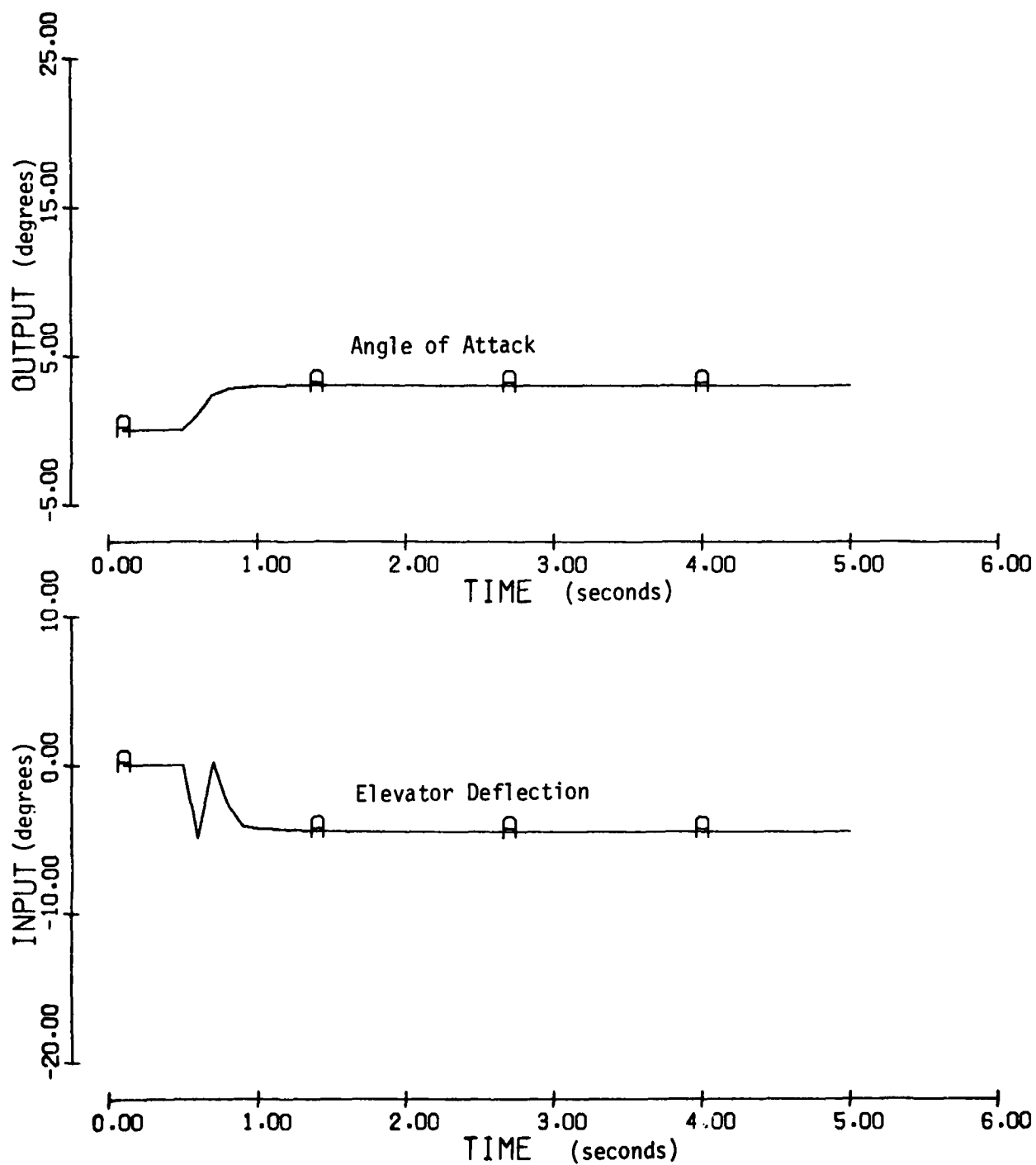
Figure 51a. Gradient algorithm, flight condition 2 baseline.

179
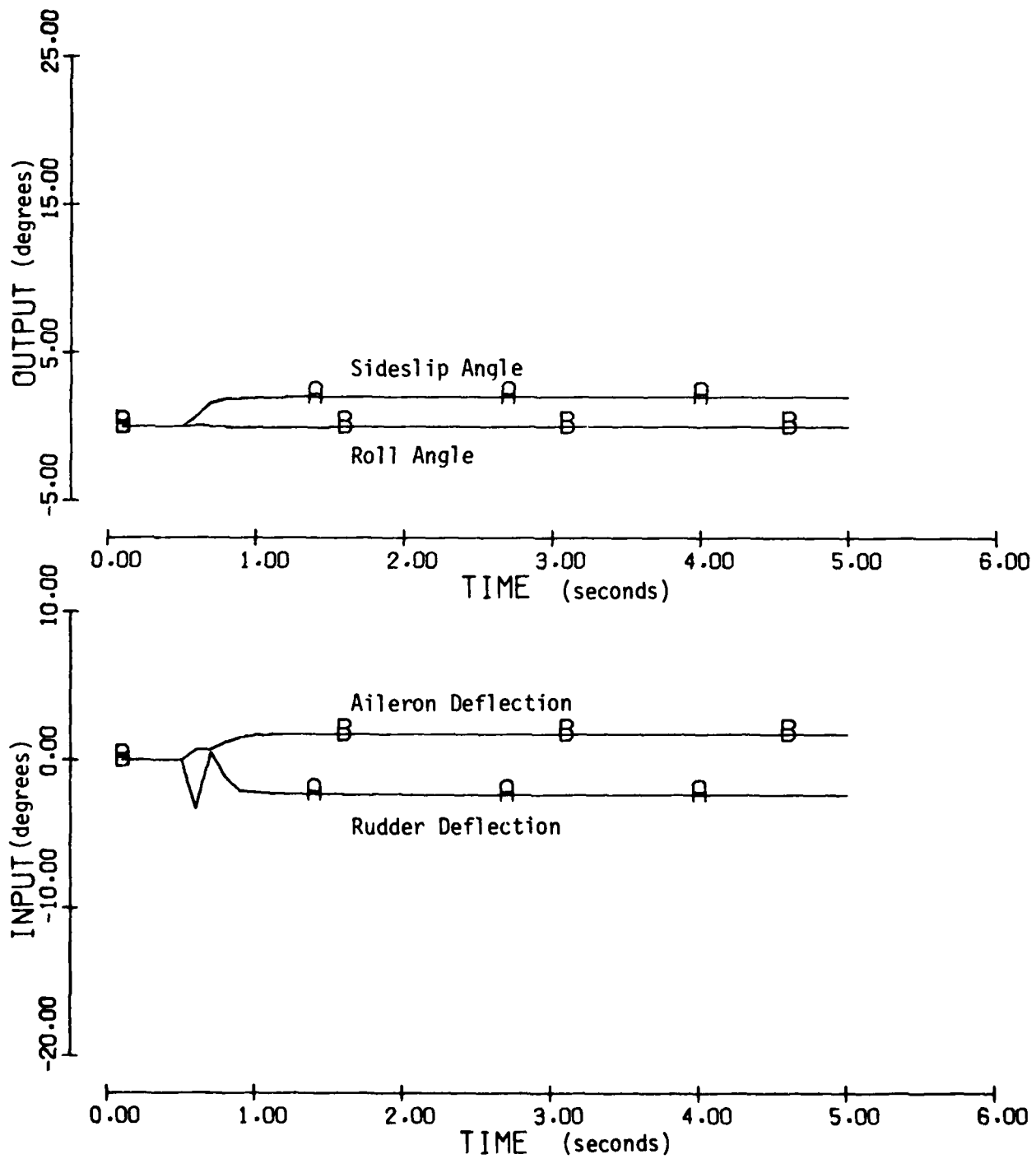
Figure 51b. Gradient algorithm, flight condition 2 baseline.

180
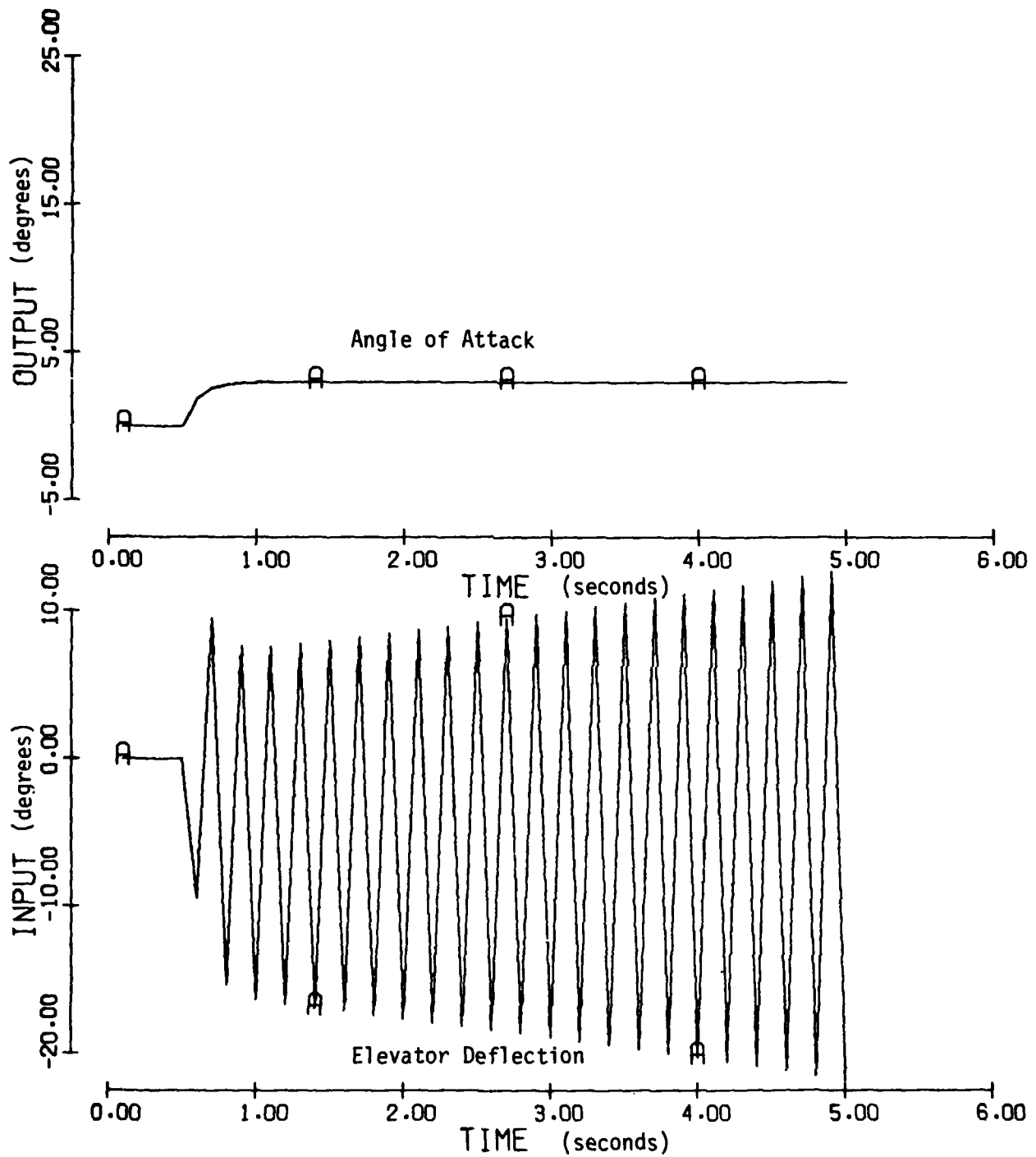
Figure 52a.   Inversion algorithm, flight condition 2 baseline.
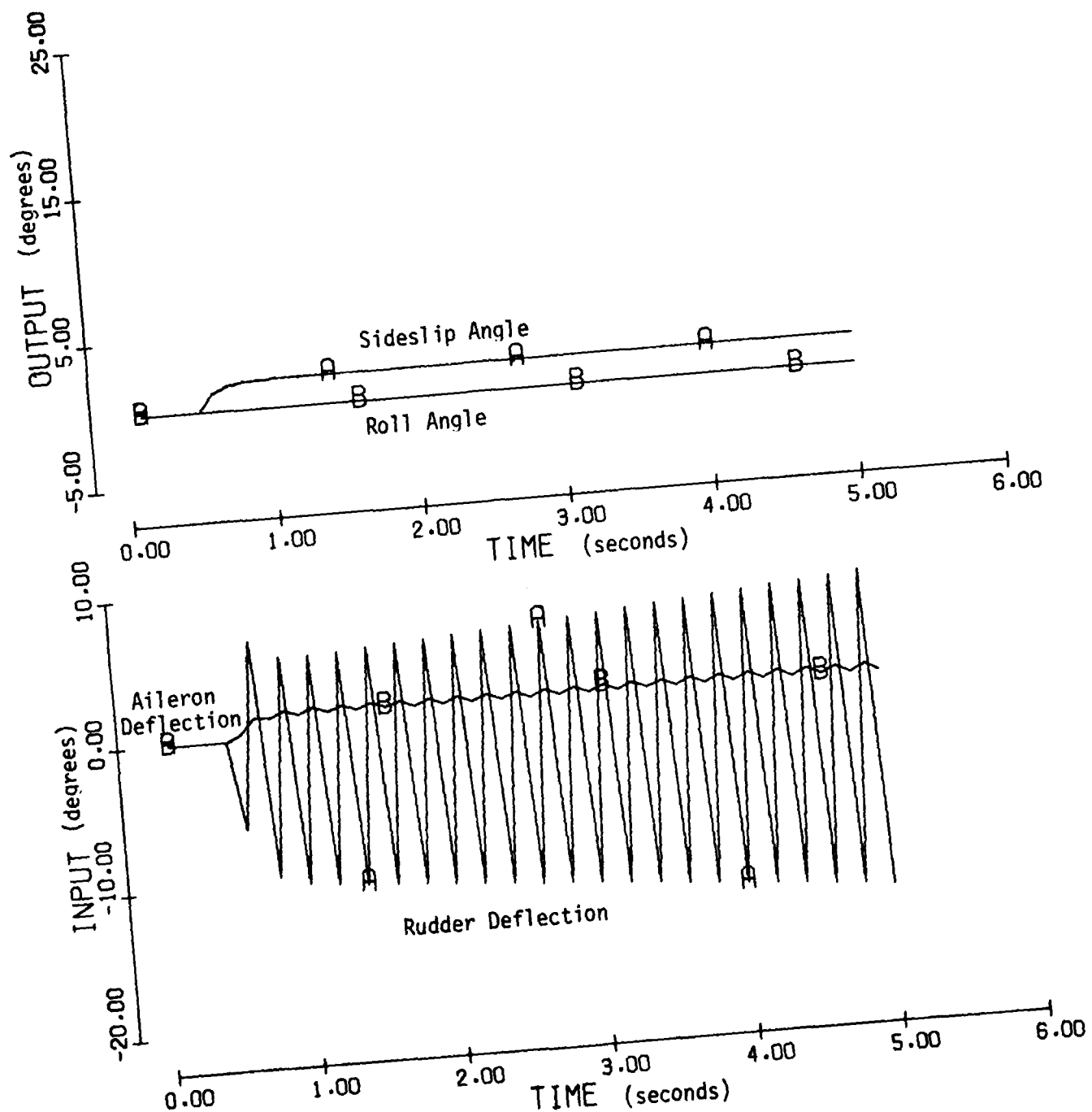
Figure 52b. Inversion algorithm, flight condition 2 baseline.

182
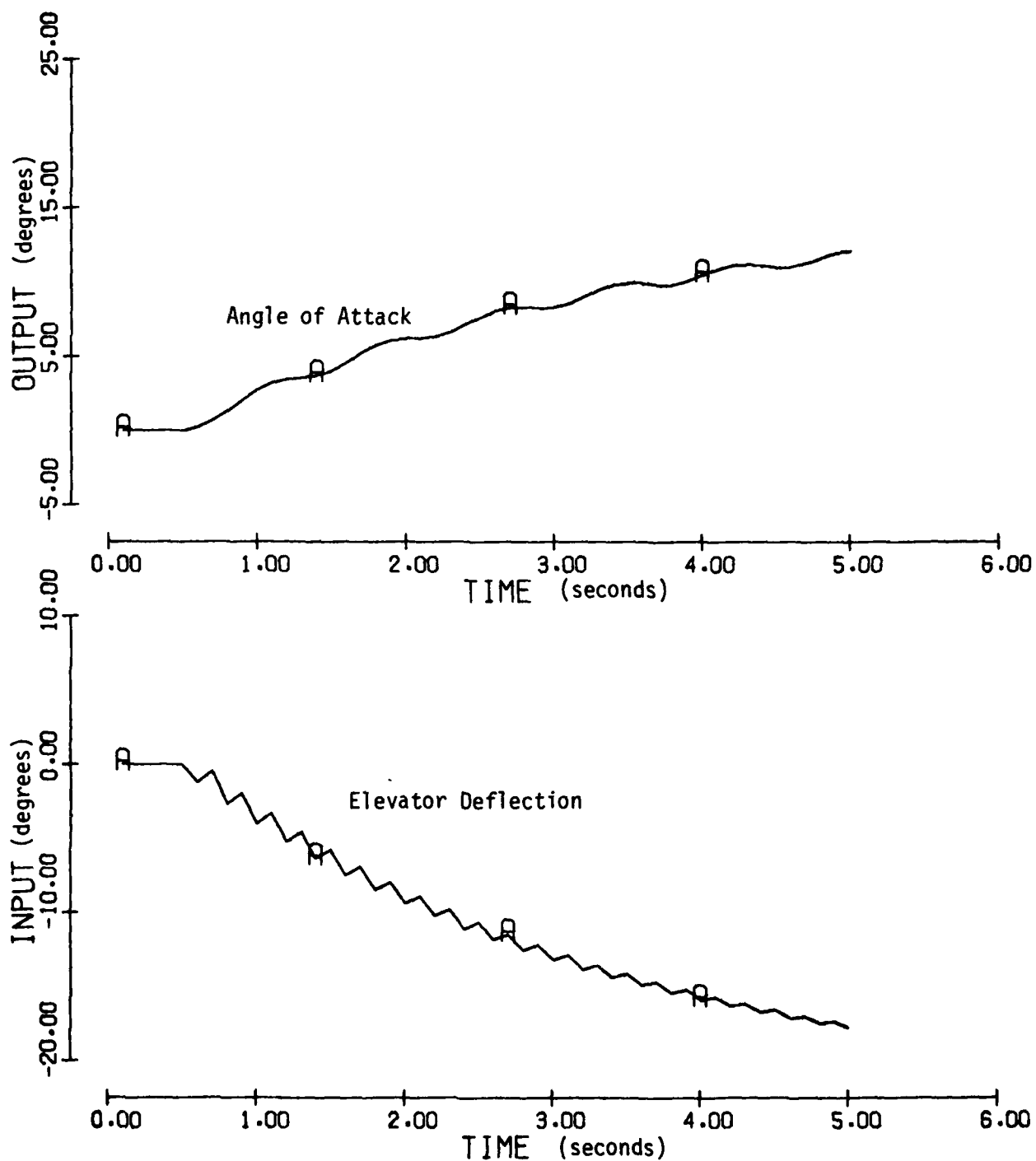
Figure 53a.   Inversion algorithm, control model from condition 1
with missile at condition 2.
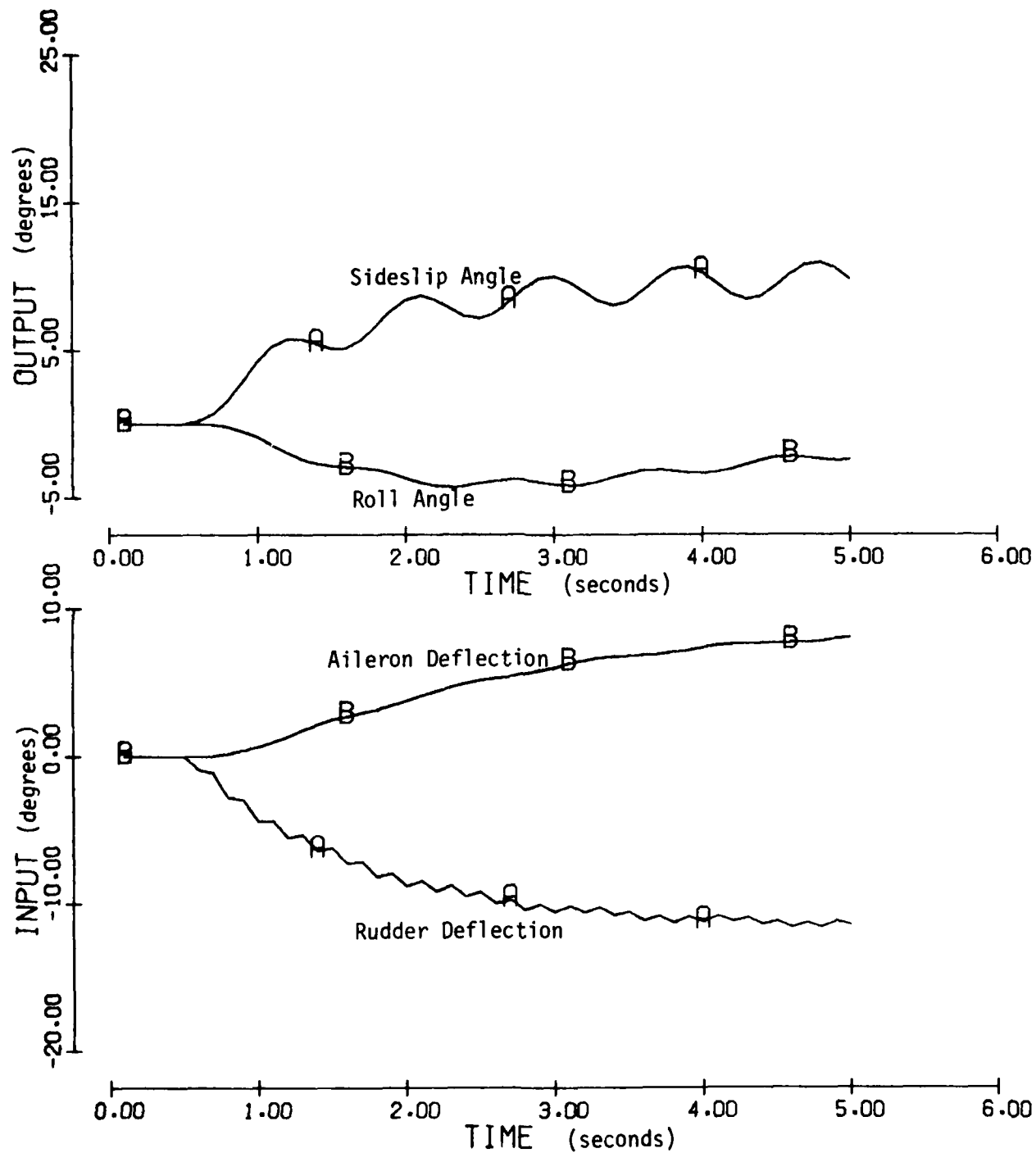
Figure 53b. Inversion algorithm, control model from condition 1 with missile at condition 2.
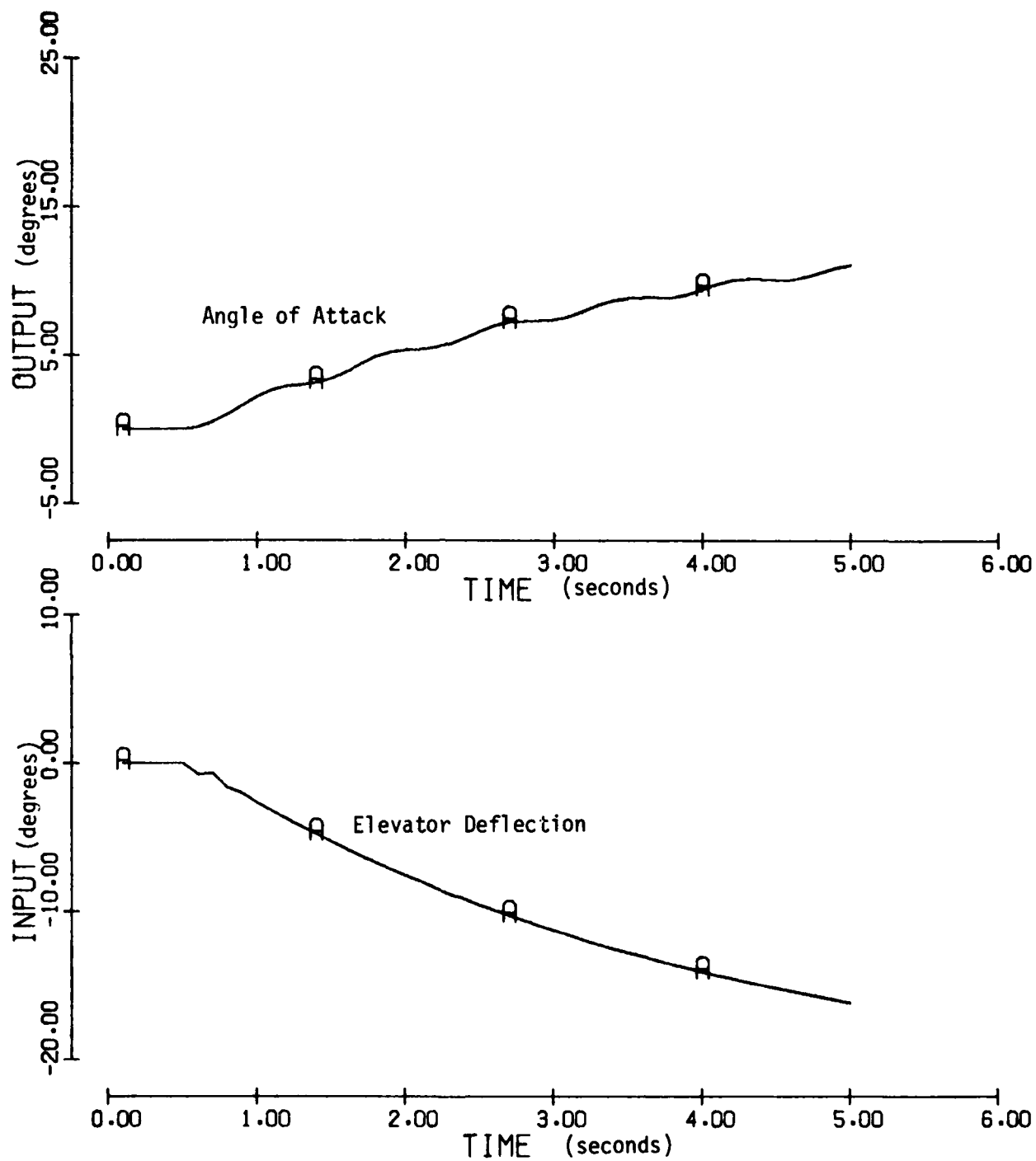
Figure 54a. Gradient algorithm, control model from condition 1 with missile at condition 2.

185
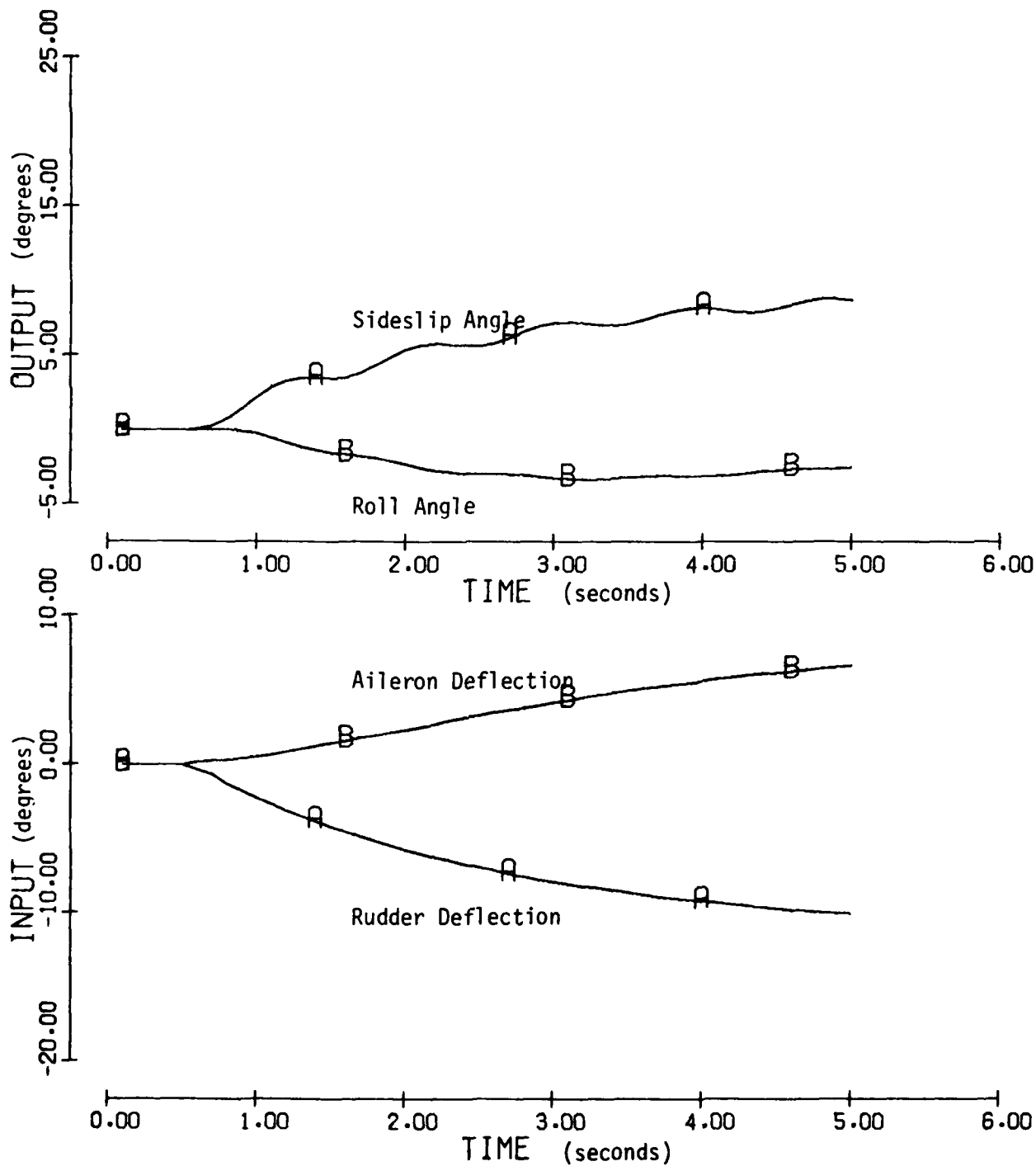
Figure 54b. Gradient algorithm, control model from condition 1
with missile at condition 2.

186
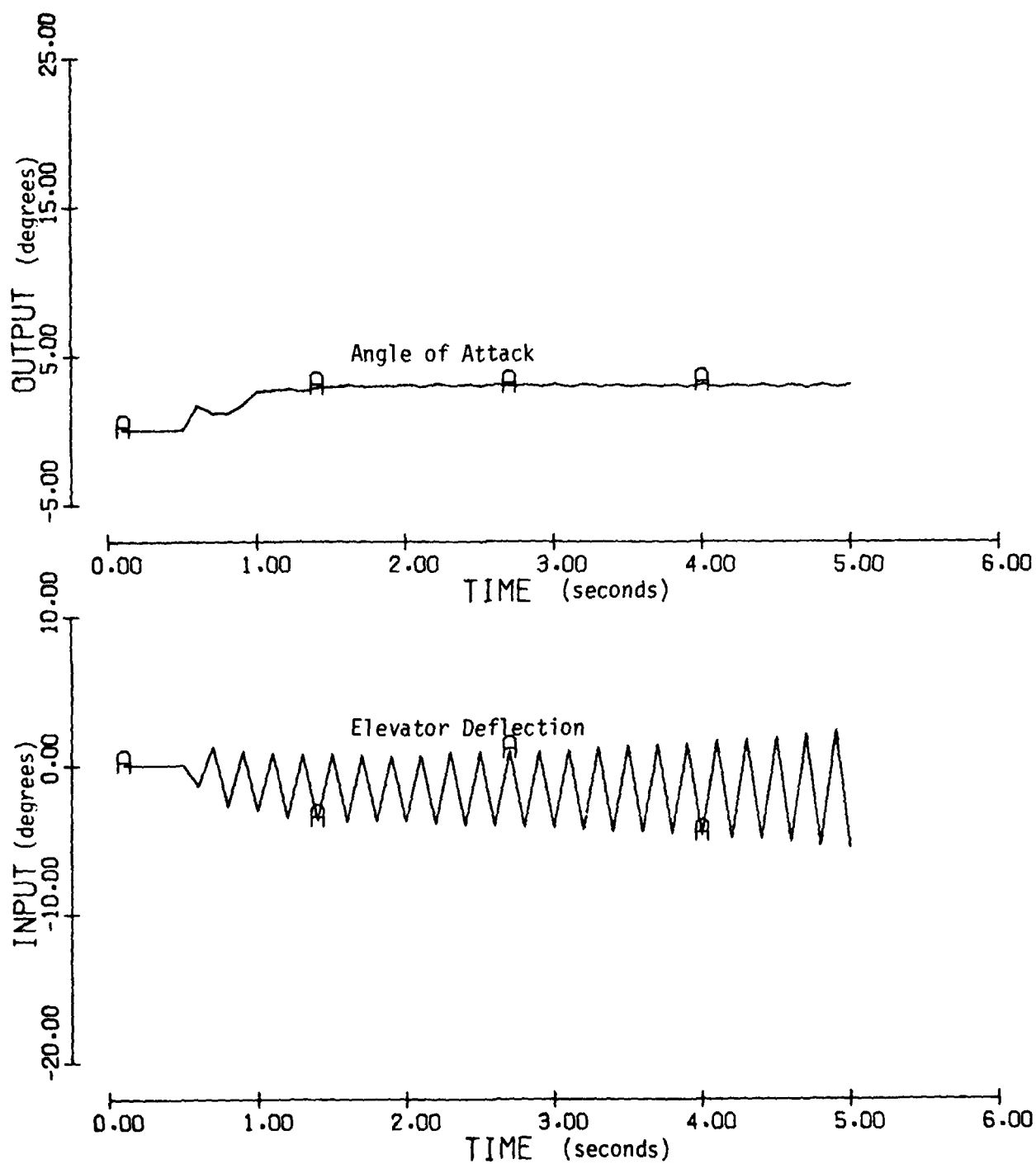
Figure 55a. Inversion algorithm, control model from condition 2 with missile at condition 1.

187
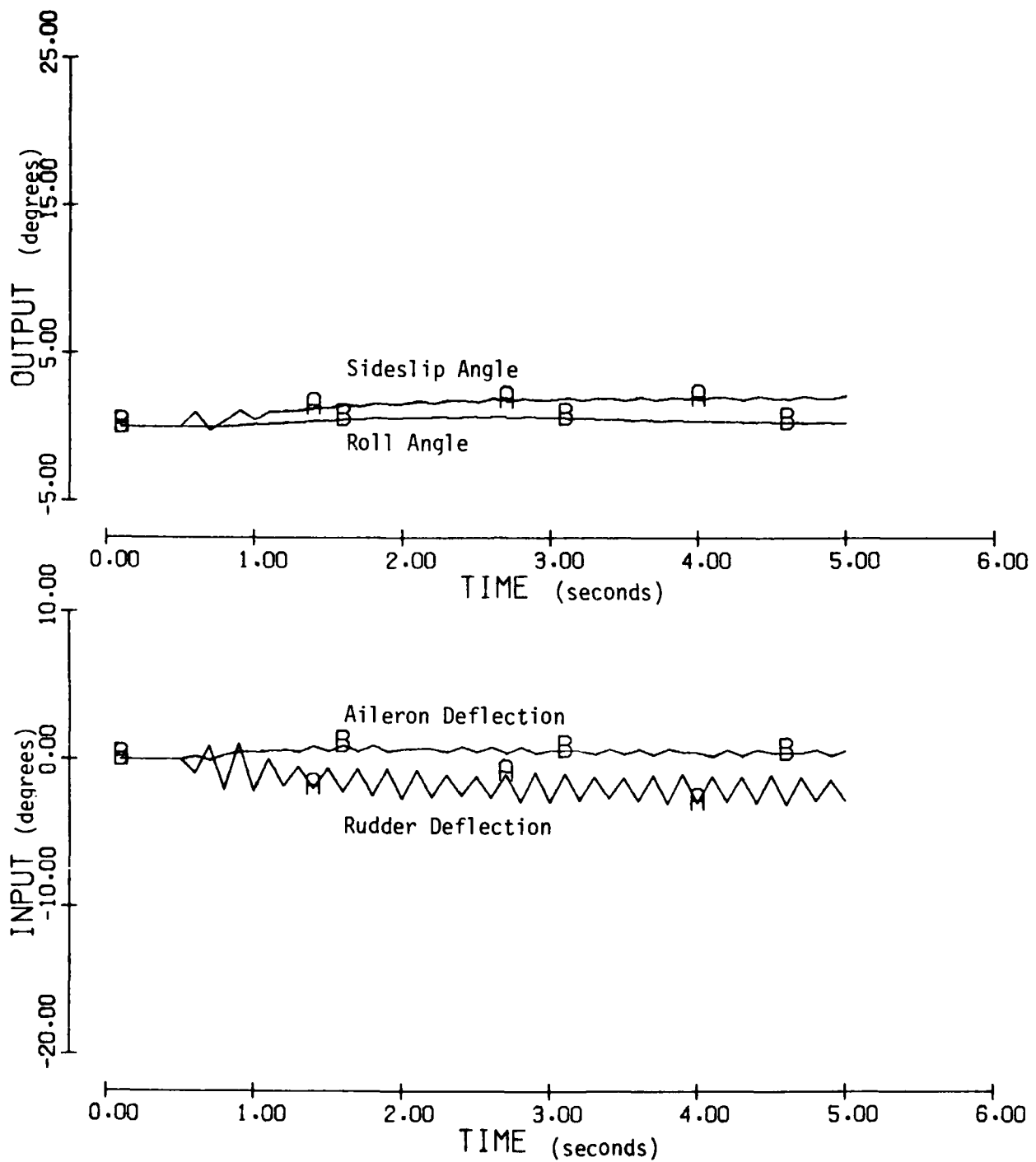
Figure 55b. Inversion algorithm, control model from condition 2 with missile at condition 1.
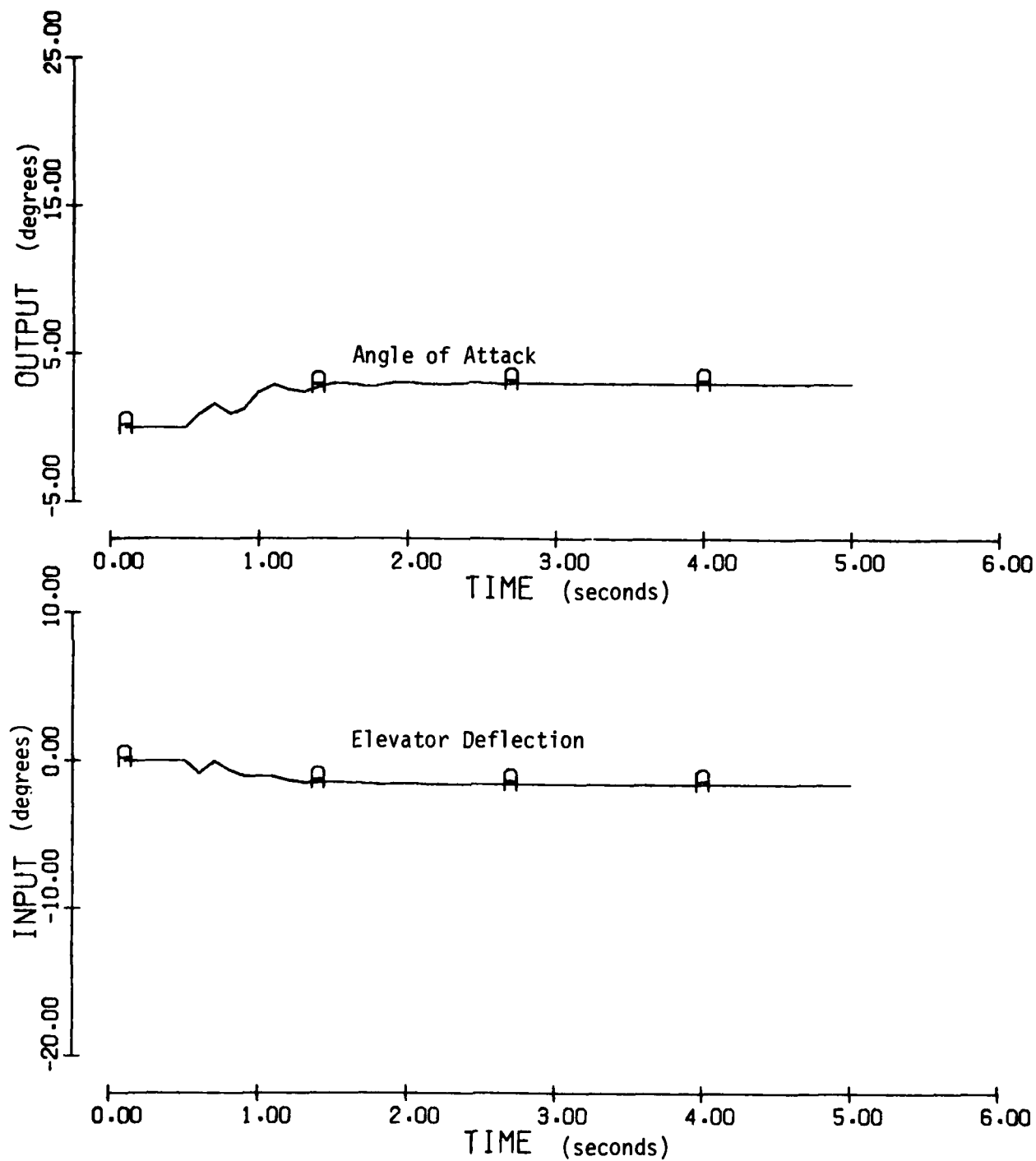
Figure 56a.   Gradient algorithm, control model from condition 2
with missile at condition 1.

189
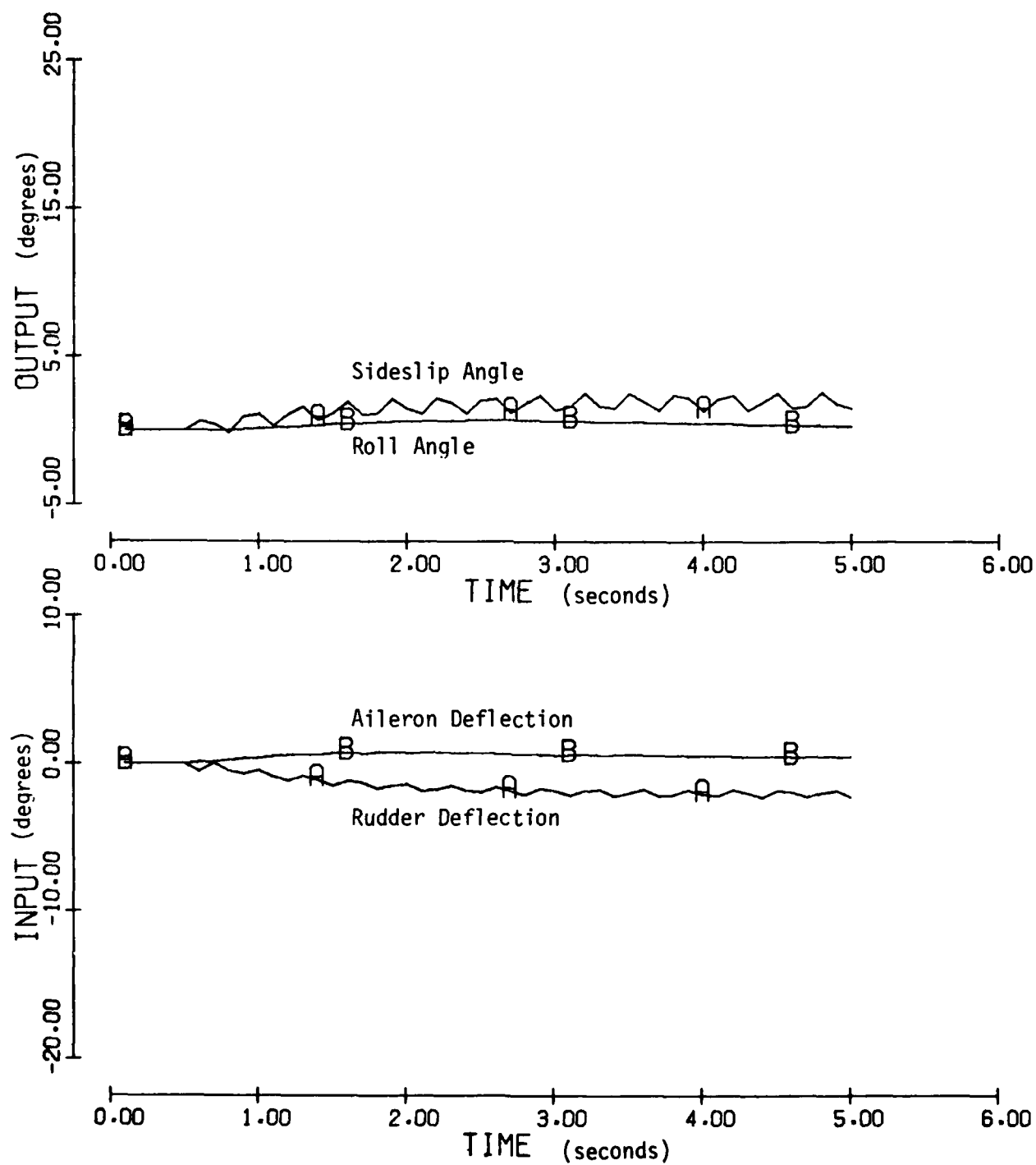
Figure 56b. Gradient algorithm, control model from condition 2
with missile at condition 1.

190

SECTION X

CONCLUSIONS AND RECOMMENDATIONS

The overall conclusion of this study is that MAC technique has a sound mathematical and empirical basis. It is a highly flexible, intuitive, and general approach to control design which fully exploits the capabilities of current microprocessors. MAC can handle hard constraints, time varying system characteristics, and unequal numbers of inputs and outputs. These features, ease of implementation and its theoretical properties of robustness make it a very powerful technique for guidance and control of flight vehicles.

Specific conclusions of this study are: (i) the mathematical properties of an idealized MAC can be analyzed by conventional control analysis techniques. For unconstrained single-input, single-output minimum phase systems, MAC design is similar to inverse control. An appropriate choice of reference trajectory is required to achieve desired robustness, tracking, and disturbance rejection properties.

(ii) Simple analytical criteria can be derived for the robustness of MAC in terms of gain margin, which agree quite well with the simulation results. It is shown that there is a direct relationship between MAC robustness and speed of response of the reference trajectory (or of the closed loop system).

(iii) In general, the performance of the closed loop prediction MAC is better than that of the open loop prediction MAC. However, there are

cases (e.g., large initial distance from the set point) where a combined use of open loop prediction and closed loop prediction strategies may be useful.

(iv) For nonminimum phase systems, inverse control leads to unbounded inputs. The MAC design approach can be easily modified to overcome this difficulty, however. The modified MAC design uses two different models: a prediction model which is a close replica of the true system and a control (or deconvolution) model which has a stable inverse. An optimal solution to the latter problem is given in terms of a Riccati equation. (The use of the Riccati equation solution, in this context, is very different from that in LQR design.) The optimal design performs much better than pole placement and weighting techniques.

(v) MAC design can be extended to continuous time systems and the optimal inputs can be computed by parameter optimization methods using suitable basis functions. The control law is shown to be a piecewise linear function of the reference trajectory and current state (or equivalently the past inputs and outputs) of the system.

(vi) For continuous time systems with discrete observations, unobserved outputs, process noise, and measurement noise, the MAC approach can be applied in conjunction with a state estimator (Kalman filter). The control computations are still performed using an impulse response model and a quadratic programming algorithm. The sample rate selection problem, however, can be studied by an extension of the LQG theory. The effect of sample rate on MAC performance can be expressed in terms of the solution to the well-known estimation and control Riccati equations.

192

(vii) The missile control simulations demonstrated superior performance achievable with MAC in a missile control environment. The basic MAC was shown to be tolerant of model mismatch and measurement noise, as indicated in the theoretical development. An alternate optimization algorithm was shown to preserve the good output tracking of the regular algorithm while using smoother control sequences. The alternate algorithm also appears to be more robust than the basic one, and just as tolerant of noise.

(viii) Control design involves subtle tradeoffs between several conflicting objectives such as tracking, robustness, constraint satisfaction and noise performance. Using the MAC approach, a large number of designs can be tried rapidly and the above tradeoff can be made easily. MAC design is also self-suggestive for improvements since it reveals the factor limiting performance.

The following recommendations are made for further research: (i) adaptive MAC: the MAC approach is easily extended to the case where the plant model is varying in an a priori unpredictable fashion. It has been shown that identification and control are dual problems in the MAC formulation. However, the theoretical properties of adaptive MAC have not been studied thoroughly. It is recommended that the convergence and robustness properties of adaptive MAC be studied analytically and the role of test signals (explicit and implicit) be quantified in terms of control performance.

(ii) Impulse response representation is not suitable for lightly damped and unstable systems. A number of heuristic techniques, including

compensation prior to control by MAC are available and have been used in practical applications. It is recommended that these techniques be analyzed theoretically to select optimum designs for specific situations.

(iii) Since the MAC approach is equally applicable to guidance design, it is recommended that both guidance and control problems for flight vehicles be investigated using the MAC approach. In particular, the application of MAC to cruise missile guidance is very promising (Reid et al., 1980).

(iv) Several algorithmic improvements in MAC design have been described in this report. It is recommended that these improvements be tested on more complex simulations.

(v) The next logical step in MAC development is testing a research aircraft such as TIFS and NAVION. This would involve determination of real-time computer requirements, comparison with other control techniques and performance evaluation.

REFERENCES

J. E. Ackermann, "A Robust Control Design," <u>Proc. Joint Automatic Control Conf.</u>, pp. 877-883, 1979.

L. V. Ahlfors, <u>Complex Analysis</u>, New York: McGraw-Hill, 1966.

Air Force Institute of Technology (AFIT), "Investigation of an Air-to Air Missile Using Ducted Rocket Technology," Wright-Patterson AFB, Ohio, Vol. 1, Final Report No. AA/78D-1, 1978.

B. D. O. Anderson and J. B. Moore, <u>Optimal Filtering</u>, Englewood Cliffs, N.J.: Prentice Hall, 1979.

K. J. Astrom, <u>Introduction to Stochastic Control Theory</u>, New York: Academic Press, 1970.

K. J. Astrom and B. Wittenmark, "Analysis of Self Tuning Regulators for Non-Minimum Phase Systems," <u>Proc. IFAC Symposium</u>, Budapest, 1974.

K. J. Astrom, "Self-Tuning Regulators--Design Principles and Applications," in <u>Applications of Adaptive Control</u> (K. L. Narendra and R. V. Monopoli, eds.), New York: Academic Press, pp. 1-68, 1980.

G. E. P. Box and G. M. Jenkins, <u>Time Series Analysis</u>, San Francisco: Holden-Day, 1976.

A. E. Bryson and Y. C. Ho, <u>Applied Optimal Control</u>, New York: John Wiley and Sons, 1975.

M. D. Canon, C. D. Callum and E. Polak, <u>Theory of Optimal Control and Mathematical Programming</u>, New York: McGraw-Hill, 1970.

A. H. Jazwinski, <u>Stochastic Processes and Filtering Theory</u>, New York: Academic Press, 1970.

H. Kwaakernak and R. Sivan, <u>Linear Optimal Control Systems</u>, New York: John Wiley and Sons, 1972.

I. D. Landau, "A Survey of Model Reference Adaptive Techniques (Theory and Applications)," <u>Automatica</u>, Vol. 10, pp. 353-379, 1974.

D. G. Luenberger, <u>Introduction to Linear and Nonlinear Programming</u>, Reading, Mass.: Addison Wesley, 1973.

R. K. Mehra, "Choice of Input Signals," in <u>Trends and Progress in System Identification</u> (P. Eykhoff, ed.), Oxford, England: Pergamon Press, 1981 (to appear).

R. K. Mehra, R. Rouhani, A. Rault and J. G. Reid, "Model Algorithmic Control: Theoretical Results on Robustness," Proc. Joint Automatic Control Conf., pp. 387-392, 1979.

R. K. Mehra et al., "Model Algorithmic Control Using IDCOM for the F100 Jet Engine Multivariable Control Design Problem," in Alternatives for Linear Multivariable Control (Sain et al., eds.), Chicago: NEC, Inc., 1978.

P. Mereau, D. Guillaume and R. K. Mehra, "Flight Control Application of Model Algorithmic Control with IDCOM (Identification and Command)," Proc. IEEE Conf. on Decision and Control, pp. 977-982, 1978.

R. W. Miller, "Asymptotic Behavior of the Kalman Filter with Exponential Aging," AIAA J., Vol. 9, pp. 537-539, 1971.

Y. Murata, Mathematics for Stability and Optimization of Economic Systems, New York: Academic Press, 1977.

K. Ogata, Modern Control Engineering, Prentice-Hall Electrical Engineering Series, Englewood Cliffs, N.J.: Prentice-Hall, 1970.

V. Peterka, "On Steady State Minimum Variance Control Strategy," Kybernetika, Vol. 8, pp. 219-238, 1972.

L. Praly, "General Study of Single Input Single Output Linear Time Variant Control Laws: Application to an Adapted Model Algorithmic Control (AMAC)," ADERSA/GERBIOS Report, 1979.

J. G. Reid, D. E. Chaffin and J. T. Silverthorn, "Output Predictive Algorithmic Control: Precision Tracking with Application to Terrain Following," Proc. Joint Automatic Control Conf., Paper FA9-F, 1980.

J. G. Reid, R. K. Mehra and E. Kirkwood, "Robustness Properties of Output Predictive Dead-Beat Control: SISO Case," Proc. IEEE Conf. on Decision and Control, pp. 307-314, 1979.

J. Richalet, A. Rault, J. L. Testud and J. Papon, "Model Predictive Heuristic Control: Applications to Industrial Processes," Automatica, Vol. 14, pp. 413-428, 1978.

A. P. Sage and C. C. White, Optimum System Control, Englewood Cliffs, N. J.: Prentice-Hall, 1977.

F. C. Schweppe, Uncertain Dynamic Systems, Englewood Cliffs, N.J.: Prentice-Hall, 1973.

S. A. Tretter, Introduction to Discrete-Time Signal Processing, New York: John Wiley and Sons, 1976.

V. I. Utkin, "Variable Structure Systems with Sliding Modes," IEEE Trans. Automatic Control, Vol. AC-22, pp. 212-222, 1977.

APPENDIX A

GENERAL STUDY OF SINGLE INPUT SINGLE OUTPUT LINEAR TIME
INVARIANT CONTROL LAWS. APPLICATION TO AN
ADAPTED MODELS ALGORITHM CONTROL (AMAC)

L. PRALY

Abstract: In this study, we come back on some characteristics of linear
time invariant control laws and we show how the single input single output
(SISO) adapted model algorithm control (AMAC) is a technique for designing
such a law.

ADERSA/GERBIOS

2 Avenue du 1er Mai
91120 Palaiseau
FRANCE

A - RETURN ON LINEAR TIME INVARIANT LAWS OF CONTROL

A-1 Introduction

Let us return on what is the problem of sampled-data control systems synthesis.

Given a mathematical model of a process, functionnal operator between an input and an output, given a set of specifications, given a method to compute a control input, the problem of synthesis may be defined as follows : find the parameters used in the computation of the control input such that the mathematical process with that input meets all the specifications. We spoke about a mathematical model of a process and not about a real physical process. We will say a law of command to be robust if it can be used on a physical process.

More precisely we call robustness the coherence between approximations of a mathematical representation of a physical process, and the sensitivity of performance criteria defined by the specifications, to variations of this representation. Let P_0 be the nominal mathematical process, the control input is designed for, if the performance criteria are continuous in P_0, we can expect the satisfaction of the specifications for any P in the vicinity of P_0. Then, the physical process we want to command must have representations each in this vicinity.

Another way to formulate the problem is : the set of mathematical processes, images of the physical process, must be enclosed in the set of mathematical processes which verify the specifications for a given control law.

So we will successively

- define a mathematical process

- define a set of specifications

- define a control law

- find relationships between parameters of the control
  law

- study the sensitivity of the performance criteria.

A-2   Definitions

A-2.1   Definition of the mathematical model of a process

A-2.1.1   Definition

We define here a discrete time mathematical model of a process as a transformation of a set of sequences called inputs into another set of sequences called ouputs.

We differentiate three types of signals between the input sequences :

a controlable measured signal called control and noted e_n

an uncontrolable but measured signal called measured disturbance, and noted v_n

an uncontrolable unmeasured signal called disturbance and noted w_n.

For single input single output systems e,v,w are scalars and so is the output noted s_n.

So, if P is an operator on sequences, we have the relation between inputs and output :

$$s(.)=P(e(.),v(.),w(.)).$$

A-2.1.2   Hypotheses

A-2.1.2.1 Hypothesis on P(H1) :

We suppose P to be a linear time invariant operator which is of rational type and asymptotically stable. Moreover we suppose a non zero static gain.

A-2.1.2.2   Hypothesis on disturbances(H2) :

We suppose both measured and unmeasured disturbances to be causal and to admit z - transforms which verify conditions of final value therorem [1].

If the disturbances are represented as stochastic processes, these hypotheses are made on the mathematical expectations and all the following deterministic results must be considered in mathematical expectation.

Moreover we suppose the output to be lineraly time-invariant dependant on the disturbances. So we introduce a new linear, time invariant, asymptotically stable operator Q between the measured disturbance and the output.

A-2.1.3   Representation of the mathematical model of the process

With hypotheses H1, H2, we compute the output s(n), from the inputs e(n),v(n),w(n) by the recursive equation :

$$\sum_{i=0}^{N_f} f_i \, s_{n-i} = \sum_{i=0}^{N_g} g_i \, e_{n-i} + \sum_{i=0}^{N_n} h_i \, v_{n-i}$$

$$+ \sum_{i=0}^{N_f} f_i \, w_{n-i} \qquad (1)$$

where $(f_i)_{i\in(0,.,N_f)}$, $(g_i)_{i\in(0,.,N_g)}$, $(h_i)_{i\in(0,.,N_h)}$ are time invariant scalars.

Neglecting the initial conditions (justified by asymptotic stability), we can represent (1) in a more concise way using z - transforms

$$s(z) = P(z)e(z) + Q(z)v(z) + w(z) \qquad (2)$$

201

where :

$$P(z) = \frac{p_n(z)}{p_d(z)}$$

$$p_n(z) = \sum_{i=0}^{N_g} g_i \, z^{N_g - i} \, ,$$

$$Q(z) = \frac{q_n(z)}{q_d(z)} \, ,$$

(3)

$$q_n(z) = \sum_{i=0}^{N_h} h_i \, z^{N_h - i} \, ,$$

$$p_d(z) . q_d(z) = \sum_{i=0}^{N_f} f_i \, z^{N_f - i} \, ,$$

and from the causality principle, degree of $p_n(q_n)$ is less than degree of $p_d(q_d)$.

Moreover from the hypotheses, the roots of $p_d(z)$ and $q_d(z)$ are strictly in the unit circle.

So we get the block representation given by figure 1 :
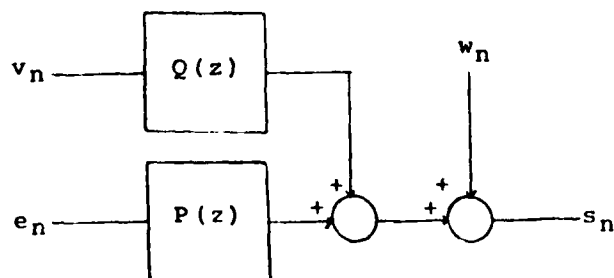
FIGURE 1 - Representation of the process

A-2.2   Definition of a set of specifications

A-2.2.1   Output regulation

We want the effect of non decreasing distur-
bances on the process to be, in some sense, minimized or
eliminated.

A-2.2.2   Output tracking

Given an external non diminishing signal
called the set point and noted u_n , we want the output
s_n to track u_n with minimal or ideally, zero steady
state error. For this problem, we impose a causal set point
with z - transform which verifies conditions of the final
value theorem.

A-2.2.3   Internal stability

In both cases it is also imperative that
an appropriate control law be designed in such a way as to
insure an asymptotically stable design i.e. the relations
between the external signals (set point, measured and
unmeasured inputs) and the internal signals (control, output)
must be stable in some sense.

A-2.2.4   Asymptotic convergence

We will summarize the preceding definitions
by the asymptotic convergence of the output s_n to the
set point u_n

$$\lim_{n \to \infty} u_n - s_n = 0 \tag{4}$$

203

In fact we have only here the least constraints to set any system of control. The synthesis of such a system must also take into account the behaviour of this convergence and need performance criteria [2]. From greater variability we keep ourselves within the convergence criterion.

A-2.3 Definition of the control law

A-2.3.1 Definition

We call control law a method to compute future controls given the observation of all the measurable past signals. To get a very general linear time-invariant control, we compute a future control e_{n+1} , given the past measured signals $(e_m,\ s_m,\ u_m,\ v_m\ ;\ m \leqslant n)$ as a finite linear combination :

$$e(n+1) = -\sum_{i=0}^{N_a} a_i\ e_{n-i} - \sum_{i=0}^{N_d} d_i\ s_{n-i} + \sum_{i=0}^{N_r} r_i\ u_{n-i} - \sum_{i=0}^{N_b} b_i v_{n-i}$$

(5)

Or using z - transforms, we write :

$$c(z)e(z) = r(z)u(z) - d(z)s(z) - b(z)v(z) \qquad (6)$$

with $c(z)$, $r(z)$, $d(z)$, $b(z)$ z - polynomials such that degree of $c(z)$ is greater than degree of $r(z)$, $d(z)$ or $b(z)$ and $c(z)$ is mutually prime with $r(z)$, $d(z)$ and $b(z)$.

Note that from the homogeneity of equation (6), there is no use to take rational functions instead of polynomials.

A-2.3.2 Interpretation

Equation (6) has the block diagram representation given in figure 2.
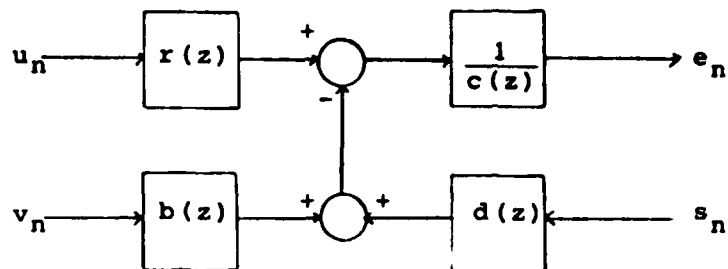
FIGURE 2 - Control law structure

So we can interpret the four parameters c,r,d,b of the control law as [3] :

c(z) is a compensator

d(z) is a sensor

r(z) is a reference

b(z) is a feed-forward input.

A-3 Relations between parameters of the control law

A-3.1 Study of the closed loop-system

We study the closed loop-system in its asymptotic behaviour. So we are going to express the various transfers between external and internal signals :

The closed loop system is represented by figure 3
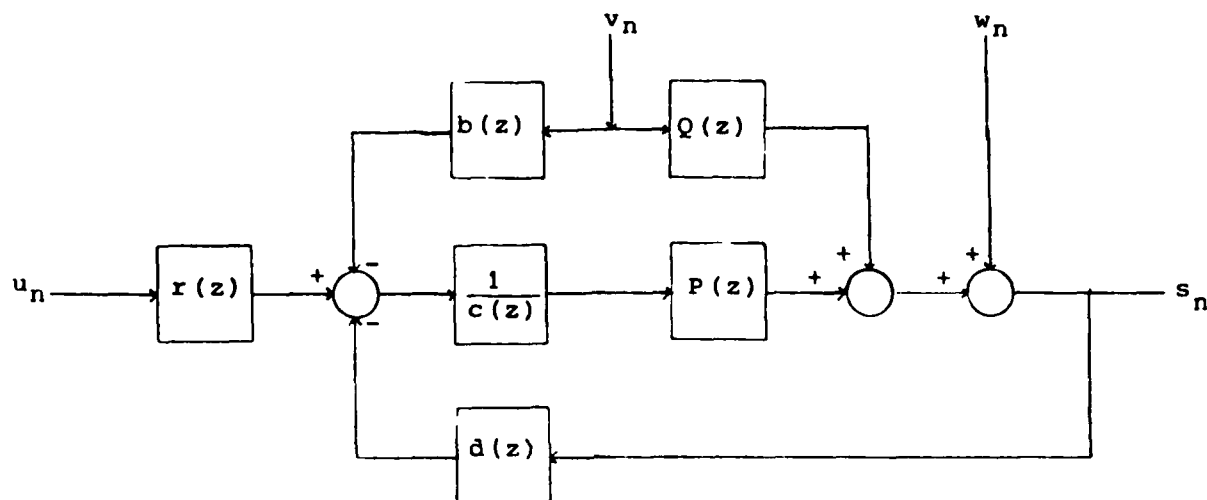
FIGURE 3 - Closed-loop system

We have the z - transform relations :

$$s(z) = S_a(z) u(z) + S_{rv}(z) v(z) + S_{rw}(z) w(z) \qquad (7)$$

$$e(z) = E_a(z) u(z) + E_{rv}(z) v(z) + E_{rw}(z) w(z) \qquad (8)$$

with : the tracking transfers

$$S_a(z) = \frac{r(z) P(z)}{c(z) + d(z) P(z)} \qquad (9)$$

$$E_a(z) = \frac{r(z)}{c(z) + d(z) P(z)} \qquad (10)$$

and the regulation feedback and feedforward transfers :

$$S_{r_v}(z) = \frac{c(z)Q(z) - b(z)P(z)}{c(z) + d(z)P(z)} \qquad (11)$$

$$S_{r_w}(z) = \frac{c(z)}{c(z) + d(z)P(z)} \qquad (12)$$

$$E_{r_v}(z) = -\frac{b(z) + d(z)Q(z)}{c(z) + d(z)P(z)} \qquad (13)$$

$$E_{r_w}(z) = -\frac{d(z)}{c(z) + d(z)P(z)} \qquad (14)$$

We can see that the poles of any transfer are given by the roots of the expression $c(z) + d(z)P(z)$. Moreover from the stability of $P(z)$, $Q(z)$ and the hypothesis of mutual primeness, a necessary and sufficient condition of internal stability is given by the stability of the control and more precisely by the stability of the $E_a(z)$ transfer.

We shall note that given the stability conditions, the sensor $d(z)$ determine the $E_{r_w}(z)$ regulation transfer, the compensator $c(z)$ determines the $S_{r_w}(z)$ regulation transfer and $r(z)$ determines the $E_a(z)$ tracking transfer. With the error tracking transfer :

$$1 - S_a(z) = S_{r_w}(z) + \frac{(d(z) - r(z))P(z)}{c(z) + d(z)P(z)} \qquad (15)$$

we remark that the difference between $d(z)$ and $r(z)$ differenciates between regulation and tracking behaviours.

Whith expression (11), if it is possible to get :

$$c(z)Q(z) = b(z)P(z) \qquad (16)$$

we will be able to compensate completely the measured disturbance.

From expressions (9), (11), (12), we can write expression (4) using the final value theorem :

$$\lim_{z \to 1_+} (1-z)(s(z)-u(z))=0 \qquad (17)$$

Supposing $s(z),u(z)$ to be defined in the ring $(1,+\infty)$. In order to separate set-point, measured and unmeasured disturbances actions, we will transpose the set of specifications into four constraints :

regulation constraints :

$$S_{r_w}(1)=\frac{c(1)}{c(1)+d(1)P(1)} = 0 \qquad (18)$$

$$S_{r_v}(1)=\frac{c(1)Q(1)-b(1)P(1)}{c(1)+d(1)P(1)} = 0 \qquad (19)$$

tracking constraint :

$$S_a(1)=1=\frac{r(1)P(1)}{c(1)+d(1)P(1)} \qquad (20)$$

And stability constraint :

The roots of $c(z) \, p_d(z)+d(z)p_n(z)$ are strictly in the unit cercle.

A-3.2   Regulation constraints

A-3.2.1   Passive regulation

We want to impose relation (18). With the hypothesis on the process and if the sensor has a non zero static gain, it is necessary and sufficient that :

$$c(1)=0 \qquad (21)$$

208

So, we must impose a factorization of the compensator in :

$$c(z) = (z-1)m(z) \qquad (22)$$

Moreover, from now on, we will write the sensor as $k\,d(z)$ with :

$$d(1)=1, \quad k \neq 0 \qquad (23)$$

A-3.2.2 Active regulation

From the preceding results, we must impose :

$$b(1)=0 \qquad (24)$$

So, we have the following factorization

$$b(z) = (z-1)n(z) \qquad (25).$$

A-3.3 Tracking constraint

We verify expression (20) if we impose identical static gains for both sensor and reference. So as in (23), from now on, we will write the reference as $k\,r(z)$ with :

$$r(1)=1, \quad k \neq 0 \qquad (26)$$

A-3.4 Remark

With expressions (22), (23), (25) and (26), (6) must be rewritten as

$$zm(z)e(z) = m(z)e(z)+(z-1)n(z)v(z)+k(r(z)u(z)-d(z)s(z)) \qquad (27)$$

209

So the control is computed in a recursive way.

A-3.5 Stability constraint

We study the polynomial :

$$(z-1)m(z)p_d(z)+kd(z)p_n(z)$$

We know already :

- $p_d(z)$ has all its roots strictly in the unit circle

- $k,d(1),p_n(1)$ are different from zero

- degree of $m(z)$ is greater than degree of $d(z)$

- degree of $p_d(z)$ is greater than degree of $p_n(z)$

with no more hypothesis on the process, we can give a sufficient condition of internal stability (Proof in Appendix 1).

If $m(z)$ has all its roots strictly in the unit circle, it exists a vicinity of zero $V(0)$ such that if k is in $V(0)-(0)$, internal asymptotic stability is ensured if and only if :

$$km(1)P(1) > 0 \quad \text{(stability condition)} \tag{28}$$

with $P(1)$ equal to the static gain of the process.

Note that from continuity, the existence of a vicinity of k can be transposed on the existence of a vicinity of $P(z)$ as we will see in a next section, and so this permits the study of robustness as it was formulated in the introduction.

210

A-3.6 Introduction of a non linearity on the control

We will extend here the results of Rouhani [4]. We introduce a non linear compensator defined as follows (figure 4).

Let y_n be the input signal of the compensator, we compute the control e_n through the expression :

$$e_{n+1} = f_n \left(\frac{1}{m_0} \left(Y_n + \sum_{i=0}^{N-1} (m_i - m_{i+1}) e_{n-i} + m_N e_{n-N} \right) \right) \qquad (29)$$

with $\quad m(z) = \sum_{i=0}^{N} m_i \ z^{N-i}$

$f_n(x)$ a real time varying function

$$y_n \longrightarrow \boxed{\dfrac{f_n(.)}{(z-1) m(z)}} \longrightarrow e_{n+1}$$
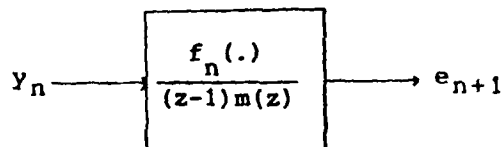
FIGURE 4 - Non linear compensator

To study the behaviour of the closed-loop system, we give an asymptotic value $\bar{u}$ to the set point, we compute a theoretic asymptotic value $\bar{e}$ of the control :

$$P(1) \bar{e} = \bar{u} \qquad (30)$$

We suppose the disturbances to be bounded and the processes to be a M.A. system $(P(z) = z^{-N_P} p_n(z))$.

211

Then we can say (Proof in Appendix 2) :

Let ρ be the greatest modulus of the roots of $(z-1)\,m(z)\,z^{N_p}+kd(z)\,p_n(z)$, if for any n we have

$$\left| f_n(x+\overline{e})-\overline{e} \right| \leqslant \frac{k}{\rho}\,|x| \qquad , \quad k < 1 \tag{31}$$

then the non linear system is stable.

In fact here (with the hypothesis on the disturbances) the stability is taken in the sense of bounded input bounded output (bibo). But if the external signals (set-point, disturbances) become constant, it will become an asymptotic stability and verify relation (4).

A-4 Sensitivity of the convergence criterion

Following our introduction we are going to study the
sensitivity of our preceding results to variations of P
and Q. In fact given the parameters m,r,k,d,m of the
control law, we are looking for the set of P,Q operators
for which the convergence criterion is satisfied. To keep
the validity of our approach, we will take P,Q in the
class of linear time invariant processes.

At once let us remark that only the stability constraint
uses hypothesis on P and Q, so we can conclude to
insensitivity of the tracking and regulation constraints.
And from now on we will look at the stability problem.

A-4.1 Sensitivity to Q

From expressions (9) to (14) it is easy to
conclude that for any asymptotically stable Q, we will have
internal stability. So, in fact, there is no sensitivity to
Q.

A-4.2 Sensitivity to P

In the hypothesis H1 we have imposed P additional
constraints to those on Q, particularly rational type and non
zero static gain. The latter was essential in regulation
and stability constraints. So we must impose variations of
P to maintain the sign of the static gain. The former was
a theoretic facility but it can be relaxed.

A-4.2.1 P of rational type

In that case we have to find all the pairs of
polynomials $(p_n(z), p_d(z))$ such that :

213

degree of p_d is greater than degree of p_n, and the roots of $p_d(z)$ and $(z-1)m(z)p_d(z)+kp_n(z)d(z)$ are strictly in the unit circle.

Given $p_d(z)$ and the number (N_p+1) of coefficients of $p_n(z)$, suppose $m(z)$ has all its roots in the unit circle, we look for the coefficients $p_0,\ldots,p_{N_p}$ such that the polynomial :

$$g(z)=(z-1)m(z)p_d(z)+kd(z)\sum_{i=0}^{N_p} p_i \, z^{N_p-i} \qquad (32)$$

has all its roots in the unit circle.

Let us work in the $g(z)$ coefficients space.

Let $\vec{G}$ be a vector representative of $g(z)$,

$\vec{M}$ be representative of $(z-1)m(z)p_d(z)$,

$\vec{P}$ be representative of $p_n(z)$,

D be a matrix representative of the action of $d(z)$ on $p(z)$.

We have :

$$\vec{G}=\vec{M}+kD.\vec{P}, \qquad (33)$$

$\vec{G}$ is linearly dependant on $\vec{P}$.

Otherwise, given the highest degree coefficient of $m(z) \, p_d(z)$, from the continuity of the coefficients on the roots, we can say that the set of admissible Gs which represent polynomials whose roots are in the unit circle is closed, bounded and connected. Moreover from the presence of $(z-1)$, we can say that $\vec{M}$ is on the frontier of this set. Thus we have the situation given by figure 5.

214
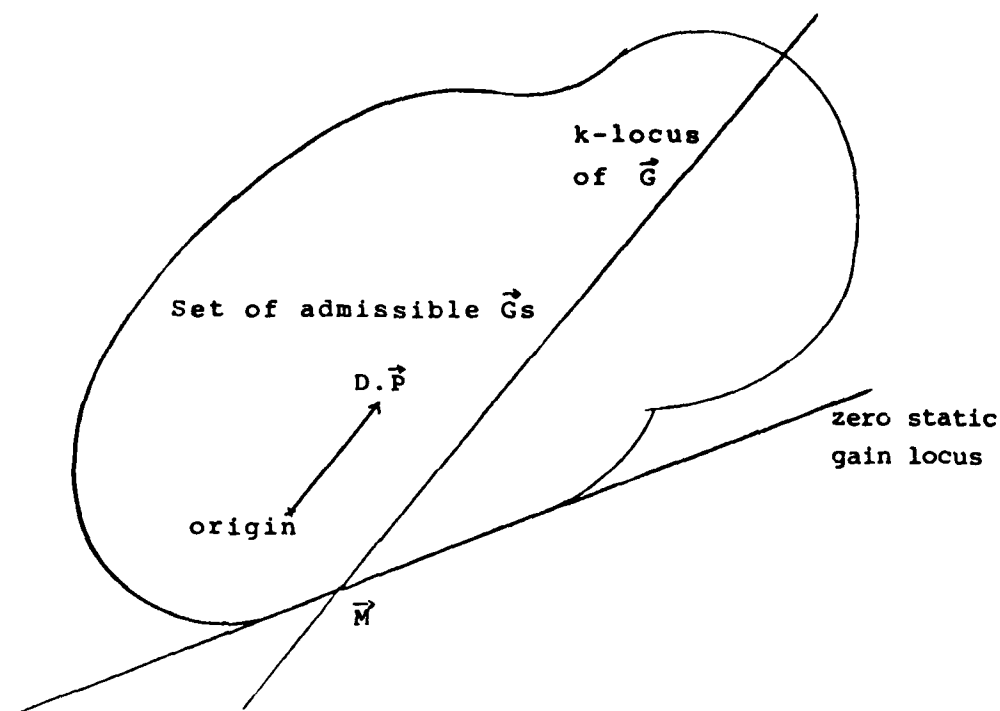
FIGURE 5 - <u>Coefficient representation of stability</u>
<u>constraint</u>

So from the knowledge of the set of admissible
$\vec{G}$s, we can find the set of admissible $\vec{P}$s. The first set has
been studied by Markov [6] in the continuous case. Particu-
larly we can't make sure of convexity of the set, so from
the linearity we don't know if the set of admissible $\vec{P}$s is
connected.

This approach gives the roles of m, d or k :
m corresponds to a translation, d is very similar to a
rotation and k to a linear displacement. Moreover we can
see the coupling between vicinities of k and P.

215

A-4.2.2 P in the vicinity of a rational P_0

Suppose the parameters of the control law to be fitted to a nominal process P_0 of rational type. We are looking for variations ΔP around P_0, such that we have internal stability.

If we suppose $P(z)$ to be an analytic function outside a domain strictly contained in the unit circle and if we note :

$$g(z) = (z-1)m(z) + kd(z)P_0(z) \qquad (34)$$

$$h(z) = kd(z)\Delta P(z) \qquad (35)$$

Then $g(\frac{1}{z})$ and $h(\frac{1}{z})$ are analytic in and on the unit circle and with the Rouché Theorem [7] , we can say that for any process $P(z) = P_0(z) + \Delta P(z)$ such that :

$$|g(e^{i\theta})| > |h(e^{i\theta})| \qquad \theta \in [-\pi, \pi] \qquad (36)$$

we will have internal stability.

We have in fact here another presentation of the result of Doyle [8] in the SISO case.

A-4.2.3 Application to a polynomial variation

Let us take $\Delta P(z)$ of the form :

$$\Delta P(z) = \sum_{j=0}^{N} \Delta p_j z^{M-j} \qquad (37)$$

Expression (36) means :

$$|(e^{i\theta}-1)m(e^{i\theta}) + kd(e^{i\theta})P_0(e^{i\theta})| > k\,|d(e^{i\theta})|\,|\Delta P(e^{i\theta})| \qquad (38)$$

$$\forall \theta \in [-\pi, \pi]$$

216

But we have with appendix 3 :

$$\left| \Delta P(e^{i\theta}) \right|^2 = \sum_{j,l=0}^{N} \Delta p_j \, \Delta p_l \, \cos(j-l)\theta \tag{39}$$

$$\left| \Delta P(e^{i\theta}) \right|^2 < \left(\sum_{j=0}^{N} \Delta p_j^2 \right) \left(\frac{N+1}{2} + \frac{1}{2} \right) \left| \frac{\sin(N+1)\theta}{\sin\theta} \right| \tag{40}$$

So we can get an upper bound of the modulus :

$$\left(\sum_{j=0}^{M} \Delta p_j^2 \right)^{\frac{1}{2}} < \min_{\theta \in [-\pi, +\pi]} \frac{\left| (e^{i\theta}-1)m(e^{i\theta}) + kd(e^{i\theta})P_0(e^{i\theta}) \right|}{k \left| d(e^{i\theta}) \right| \cdot \left(\frac{N+1}{2} + \frac{1}{2} \left| \frac{\sin(N+1)\theta}{\sin\theta} \right| \right)^{\frac{1}{2}}} \tag{41}$$

Practically an FFT algorithm will provide all these spectra.

A-4.2.4   Convergence criterion sensitivity index (CCI)

Given a process P_0 and the parameters of the control law we define an absolute index by :

$$CCI(P_0, m, k, d) = \min_{\theta \in [-\pi, +\pi]} \left| \frac{(e^{i\theta}-1)m(e^{i\theta})}{k \, d(e^{i\theta})} + P_0(e^{i\theta}) \right| \tag{42}$$

From expression (36) this index gives an upper bound on the possible spectrum variation to verify convergence criterion. So we call it a convergence criterion sensitivity index. To insure robustness, it has to be compared with an equivalent approximation index given by the P_0 model estimation phase.

A-5 Summary

In this first part, we came back on the problem of
linear control. The most important results have been
reformulated in a very general way : results on the structure
of the control law, results on stability in the linear case
and in a simple non linear case and at last results on a
measure of the sensitivity of stability.

B. THE SINGLE INPUT-SINGLE OUTPUT (SISO)
ADAPTED MODEL ALGORITHM CONTROL (AMAC)

We have just presented a linear time invariant control law in a general fashion. It is an abstract approach which serves only to ensure the convergence criterion. In an attempt to get behavior criteria, we are going to give a physical presentation through a SISO control based on the mathematical representation of the process of paragraph A-2.1.3:

$$s(z) = P(z)e(z) + Q(z)v(z) + w(z) \tag{1}$$

and the use of adapted models of the operators P, Q.

B-1. General SISO AMAC Presentation

B-1.1 Definition of the Strategy

At time n, given the past measurable signals, the SISO AMAC computes a control such that a predicted output of the process is identical to a predicted set point.

Taking the notation of Box and Jenkins [9], we write this:

$$s_n(1) = u_n(1) \tag{2}$$

the prediction being here of one point ahead. From the representation of the process (1) we decompose the predicted output into two parts: a deterministic part which functionally depends on the inputs and a non-deterministic part $su_n(1)$ resulting from the disturbances. Let $M(e_n(1),$ $e_\ell;\ \ell \leq n)$ be a model of the operator P which defines the deterministic

output from the future and past controls, we deduce from (2) the control law as:

$$M(e_n(1), e_\ell; \ell \leq n) = u_n(1) - su_n(1) \qquad (3)$$

Then to compute the control $e_n(1)$ we have three different problems: inversion of M, estimation and prediction of the non-deterministic output, and prediction of the set point.

B-1.2   Inversion of the Model M

From its definition, M is a model of the process. Note that to compute $e_n(1)$, we use this model in a reversed way compared to the physical transfer, so we require M to be invertible in the sense defined by Box and Jenkins [9] and we call it a deconvolution model. Thus with the linear time invariant hypothesis the model of the process is taken linear, time invariant, asymptotically stable, of rational type and invertible.

Let md_i or $Md(z)$ be the impulse response and the rational z-transform of this deconvolution model. We obtain from (3)

$$md_0 \cdot e_n(1) = - \sum_{i=1}^{\infty} md_i \cdot e_{n+1-i} + u_n(1) - su_n(1) \qquad (4)$$

and md_0 must be different from zero.

B-1.3   Estimation and Prediction of the Non-Deterministic Output

From expression (1) the non-deterministic output is the sum of both a filtered measured disturbance v_n and an unmeasured disturbance w_n. Suppose we have an estimation $\hat{w}_n$ of w_n and a convolution model of the

measured disturbance filter Q : $nc_i(Nc(z))$, then if $v_n(1)$ and $\hat{w}_n(1)$ are predictions of measured and unmeasured disturbances, we compute:

$$su_n(1) - nc_0 v_n(1) + \sum_{i=1}^{\infty} nc_i \cdot v_{n+1-i} + \hat{w}_n(1) \tag{5}$$

So we first need the estimation $w_n(1)$ of the unmeasured disturbance and secondly measured and unmeasured disturbance predictors.

We already introduced a convolution model Nc of Q. Let us take also a new model $mc_i(Mc(z))$ of the process P. This time we need a model to be used in the same way as the process so $Mc(z)$ is a convolution model compared with $Md(z)$, a deconvolution model. Similarly to expression (1), we compute the estimation $\hat{w}_n$ by

$$\hat{w}_n = s_n - \sum_{i=0}^{\infty} mc_i \cdot e_{n-i} - \sum_{i=0}^{\infty} nc_i \cdot v_{n-i}$$

Now from the past v_n and $\hat{w}_n$, we want to predict $su_n(1)$. From discrete parameter prediction theory [10], $v_n(1)$ and $\hat{w}_n(1)$ can be computed with prediction filters. Using z-transforms they may be expressed as

$$\hat{w}_n(1)(z) = F_w(z) \; \hat{w}(z) = \frac{fwn(z)}{fwd(z)} \; \hat{w}(z)$$

$$v(1)(z) = F_v(z) \; v(z) = \frac{fvn(z)}{fvd(z)} \; v(z) \tag{7}$$

where $fwn(z)$, $fwd(z)$, $fvn(z)$, $fvd(z)$ are polynomials in z, the degree of fwd(resp fvd) being greater than the degree of fwn(resp fvn). Moreover, to be able to predict the continuous component of the disturbances, we

impose unit static gain predictors.

Thus we get the z-transform of $su_n(1)$:

$$su(1)(z) = Nc(z) \ v(z) + Fw(z) \ \hat{w}(z) \qquad (8)$$

with $Nc'(z)$ (nc_i') computed from N_c and F_v through the relation:

$$nc_0 \ v_n(1) + \sum_{i=1}^{\infty} nc_i \ v_{n+1-i} = \sum_{i=0}^{\infty} nc_i' \ v_{n-i} \qquad (9)$$

Now from z-transform of (6) we have the final relation:

$$su(1)(z) = F_w(z)(s(z) - Mc(z)e(z)) + (N_c'(z) - F_w(z)N_c(z))v(z) \qquad (10)$$

or equivalently in the time domain:

$$su_n(1) = fw_i*(s_i - mc_i*e_i) + (nc_i' - fw_i*nc_i)*v_i \qquad (11)$$

where * represents the discrete convolution operator.

B-1.3   Set Point Prediction

To get a better behavior of the closed-loop system, at time n we need future set points. In some cases they are available (particularly when there is a hierarchical control). But generally we need a predictor. Let us take it in the form of a rational filter $Fu(z)$ with fu_i as impulse response and with unit static gain.

$$\hat{u}(1)(z) = Fu(z)u(z) = \frac{fun(z)}{fud(z)} \ u(z) \qquad (12)$$

or

$$\hat{u}_n(1) = fu_i * u_i \tag{13}$$

with fun(z), fud(z) polynomials in z, with the degree of fud(z) greater than or equal to the degree of fun(z).

B-1.4 Expression and Properties of the SISO AMAC Law

From expressions (3), (11) and (13) we get the SISO AMAC law

$$e_n(1) = \frac{1}{md_0} \left[(fw_i * mc_i - md_i) * e_i + fu_i * u_i - fw_i * s_i - (nc'_i - fw_i * nc_i) * v_i \right] \tag{14}$$

This prediction is used as the future control e_{n+1}. Thus we get the z-transform representation of the SISO AMAC law:

$$(z \cdot Md(z) - Fw(z) \cdot Mc(z))e(z) = Fu(z) \cdot u(z) - Fw(z)s(z)$$

$$- (Nc'(z) - Fw(z)Nc(z))v(z) \tag{15}$$

Then we find the expression of the four polynomials of our general linear time invariant control law:

$$c(z) = z \cdot Md(z) - Fw(z)Mc(z)$$

$$r(z) = Fu(z)$$

$$d(z) = Fw(z) \tag{16}$$

$$b(z) = Nc'(z) - Fw(z)Nc(z)$$

Thus we can give a *physical interpretation* to these polynomials. Moreover, we see that from a physical point of view c, r, d and b are not mutually independent, but Md or Mc, Nc, Fu, Fv and Fw are.

223

The regulation and tracking constraints (see Part A) are verified since we have imposed:

$$Md(1) = Mc(1)$$
$$Fu(1) = Fv(1) = Fw(1) = 1 \tag{17}$$
$$Nc'(1) = Nc(1)$$

The stability constraint (Appendix 4) can be verified by a modification of the dynamic of the unmeasured disturbance predictor if: the different models and predictors are stable, the numerator of the deconvolution model has all its roots strictly in the unit circle, the following inequality is satisfied: $Md(1) \cdot P(1) > 0$. (18)

Now assuming a perfect knowledge of the process and the measured disturbance filter:

$$Mc(z) = P(z); \quad Nc(z) = Q(z) \tag{19}$$

we can write the expected closed-loop tracking and regulation transfer:

$$S_a(z) = \frac{Fu(z)P(z)}{z\,Md(z)} \tag{20}$$

$$S_{rw}(z) = 1 - \frac{Fw(z)Mc(z)}{z\,Md(z)} \tag{21}$$

$$S_{rv}(z) = Q(z) - \frac{Nc'(z)Mc(z)}{z\,Md(z)} \tag{22}$$

So the closed-loop tracking transfer is the product of the set point predictor and the deconvolution model mismatch of the process. Similarly we get the closed-loop regulation transfer. Thus in a perfect matching, the various predictors specify the tracking and regulation closed-loop transfers.

The block representation of the SISO AMAC is given by Figure 6.
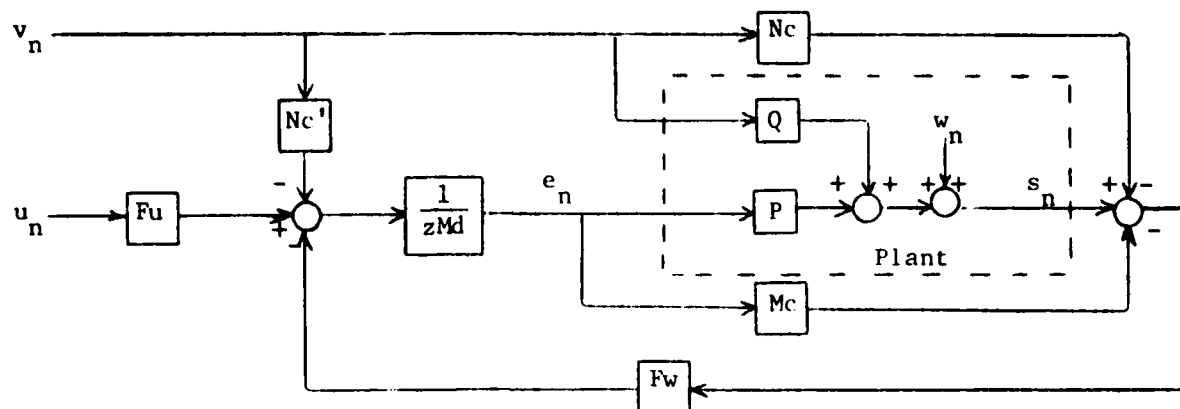
Figure 6.   SISO AMAC Representation

Mc   convolution model of the process

Md   deconvolution model of the process

P   deterministic part of the plant

Q   stochastic part of the plant (disturbance process)

Nc   convolution model of the disturbance process

Nc' predictive convolution model of the disburbance process

Fu   set point predictor

Fw   unmeasured disturbances (w_n) predictor

B-2. SISO AMAC Examples

Following our presentation we will present two classical control systems used whenever the convolution and deconvolution model can be identical.

225

B-2.1   The Optimum Control System of Phillipson [11]

Let us suppose no measured disturbance and an asymptotically stable process with a delay of ℓ samples. We take a convolution model with a delay of k samples, k underestimation of ℓ:

$$Mc(z) = z^k Md(z) \qquad (23)$$

with $Md(z)$ supposed to have all its roots strictly in the unit circle. Then if we take a unit gain element as a set point predictor, and a k-step-predictor $z^k H(z)$ for the disturbance, we obtain the optimal control system of Phillipson (Figure 7) which is an improvement over the Smith controller.
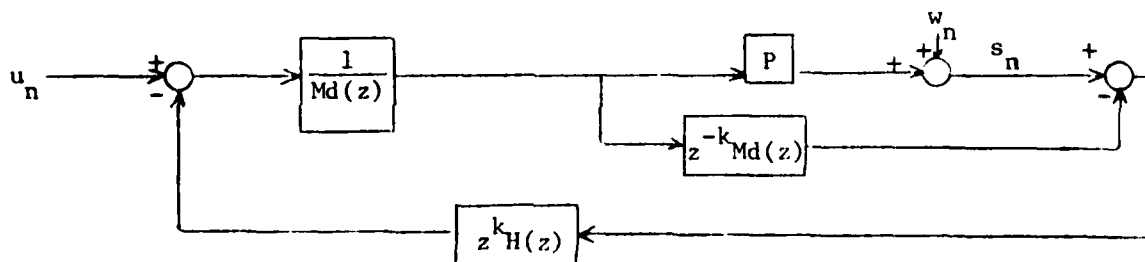
Figure 7.   Optimum Control System of Phillipson

As mentioned by Phillipson, this system used in regulation is equivalent to the Box-Jenkins-Astrom minimum-variance control [9] or to the Kalman linear regulator [12].

Thus, this system is essentially made for regulation. Moreover, the use of the inverse model as controller since this will be a high-pass filter, amplifies noise, causes violent changes in the control signal and perhaps frequent saturation. That is why the AMAC uses here an adapted model and

226

avoids rapid changes in set point thanks to the set point predictor.
Predictor H can be easily computed when the disturbance can be described
as the output of a known rational filter whose input is an independent
zero-mean random sequence. But to verify internal stability we must
not forget the constraints on the denominator of H. Here Phillipson
suggests the use of exponential smoothing for prediction to solve the
problem. That way, we can answer satisfactorily the output regulation
but not so properly the output tracking. The model predictive heuristic
control which follows attempts to answer the two questions introducing
a set point predictor and deducing the disturbance predictor.

B-2.2   Model Predictive Heuristic Control (MPHC) [13]

We give here a simplified study of this method; the very general
study can be found in [14]. Suppose no measured disturbance (MPHC can
be extended to this case) and a convolution model having a moving average
(MA) structure with all its roots strictly in the unit circle, we take:

$$M(z) = Mc(z) = Md(z) \qquad (24)$$

For the set point and disturbance predictors, we choose

$$F_u(z) = \frac{1 - G(1)}{1 - z^{-1}G(z)}$$

$$\qquad (25)$$

$$F_w(z) = \frac{1 - G(z)}{1 - z^{-1}G(z)}$$

where C(z) is a nonzero static gain transfer such that Fu(z) and Fw(z)
satisfy stability conditions.

Then from (15) we get the MPHC law:

$$(z - \frac{1 - G(z)}{1 - z^{-1}G(z)}) \cdot M(z) \cdot e(z) = \frac{1}{1 - z^{-1}G(z)} [(1 - G(1))u(z) - (1 - G(z))s(z)]$$

(25)

$$(z-1) \cdot M(z) \cdot e(z) + s(z) = (1-G(1))u(z) + G(z)s(z) \tag{26}$$

Let us develop the strategy of this relation.

Both terms are similar to outputs. We call the left-hand term a predicted output $s_P(z)$ and the right-hand term a reference output $s_R(z)$. From

$$s_P(z) = zM(z)e(z) + (s(z) - M(z)e(z)) \tag{27}$$

we define the predicted output as the output of the model at time (n+1) corrected of the estimation $\hat{w}(n)$ of the disturbance

$$s_P(n+1) = s_M(n+1) + \hat{w}(n) \tag{28}$$

with $s_M(n+1)$ output of the model with a predicted input $e_n(1)$. The reference output $s_R(z)$ is given by a trajectory connecting the past outputs of the process to the present set point.

$$s_R(n+1) = (1 - \sum_{i=0}^{\infty} g_i)u_n + \sum_{i=0}^{\infty} g_i s_{n-i} \tag{29}$$

Thus the MPHC strategy consists in computing future inputs such that predicted outputs are on a connecting trajectory. Its block representation is given by Figure 8.
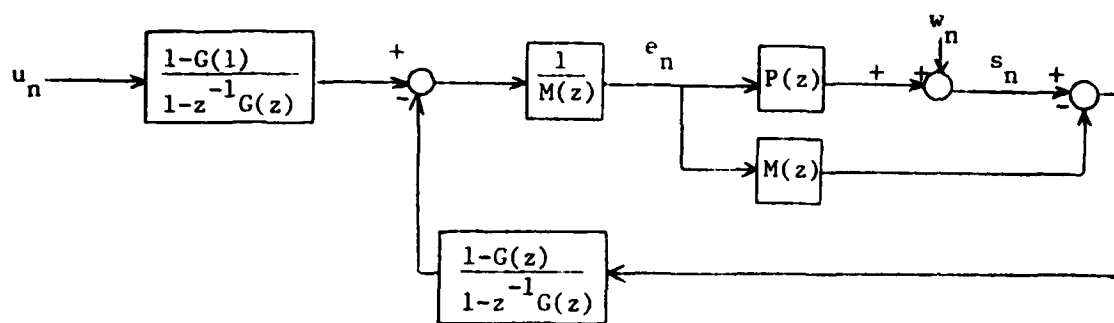
Figure 8. MPHC Representation with G(z) as Connecting Trajectory Generator

From part A, we will satisfy the convergence criterion if M(z) has all

its roots strictly in the unit circle and (1-G(1)) is taken as the stability

coefficient. But again, the transfers are not independent, for instance in a

perfect modeling we have:

$$S_a(z) = \frac{1 - G(1)}{z - G(z)} \tag{30}$$

$$S_{rw}(z) = 1 - \frac{1 - G(z)}{z - G(z)} = \frac{z-1}{z-G(z)} = \frac{z-1}{1-G(1)} \; S_a(z) \tag{31}$$

The regulation transfer is the discrete differentiation of the tracking

transfer. Moreover, if the model does not verify the stability condition,

the strategy must be seriously questioned but it has been extended to this

case by the introduction of the notion of adapted model [14].

B-3. Choice of the Deconvolution Model

We have seen that the most general linear time invariant control law

contains five independent physical components. Theoretically each can be

obtained from a modeling (system or spectrum). But the deconvolution

model is a special case because its use is not a physical one. We are

229

going to show where the problem is and how to solve it.

B-3.1 Terms of the Problem and Mathematical Solution

Let Md(z) be this deconvolution model

$$Md(z) = \frac{mdn(z)}{mdd(z)} \tag{32}$$

with mdn(z), mdd(z) polynomials in z such that with expectation (4) degree of mdd(z) is equal to degree of mdn(z).

Mc(z) is the knowledge of the process, i.e., the convolution model:

$$Mc(z) = \frac{mcn(z)}{mcd(z)} \tag{33}$$

with mcn(z), mcd(z) polynomials in z, with degree of mcd(z) greater than degree of mcn(z). The differences between these models are in their use. Let e,s be the input and the output, we write

$$s(z) = \frac{mcn(z)}{mcd(z)} \, e(z) \tag{34}$$

similarly to the process, but:

$$e(z) = \frac{mdd(z)}{mdn(z)} \, s(z) \tag{35}$$

is a reversed relation compared with the process.

As we want a stationary control law, following Box and Jenkins [9], we have to improve the stability of both transfers $\frac{mcn(z)}{mcd(z)}$ and $\frac{mdd(z)}{mcn(z)}$. The former can be ensured from the stability of the process. But the

230

latter has no physical significance and we have seen that we must impose mcn(z) to have all its roots strictly in the unit circle.

As mdd(z) has no constraint in the deconvolution use, we can take:

$$md(z) = mdd(z) = mcd(z) \qquad (36)$$

Moreover, to get a zero static gain compensator, with expression (16), we must impose:

$$mdn(1) = mcn(1) \qquad (37)$$

Thus the problem is: knowing the model of the process mcn(z), how to choose mdn(z) such that it keeps the significance of a model and it satisfies the stability condition.

If mcn(z) has all its roots strictly in the unit circle, we take obviously:

$$mdn(z) = mcn(z) \qquad (38)$$

So the real problem occurs when mcn(z) has roots on both sides of the unit circle. Let us factorize mcn(z) into:

$$mcn(z) = min(z) \cdot mon(z) \qquad (39)$$

where min(z) has all its roots strictly inside the unit circle, mon(z) has all its roots strictly outside the unit circle. We don't deal with unit modulus roots. As mdn(z) is used as a denominator, let us consider:

$$mid(z) = \frac{md(z)}{min(z)} \qquad (40)$$

$$\text{mod}(z) = \frac{1}{\text{mon}(z)} \tag{40}$$

mid(z) is a holomorphic function defined outside a domain strictly contained in the unit circle, so its Laurent expansion in the vicinity of the unit circle is:

$$\text{mid}(z) = \sum_{j=0}^{\infty} \text{mid}_j z^{-j} \tag{41}$$

mid(z) corresponds to a causal impulse response and so has a physical significance.

Inversely, mod(z) is a holomorphic function defined in a domain strictly containing the unit circle, so its Laurent expansion in the vicinity of the unit circle is

$$\text{mod}(z) = \sum_{j=0}^{\infty} \text{mod}_j z^j \tag{42}$$

Thus, mod(z) can be considered as corresponding to a noncausal impulse response. And so expression (35) or (3) implies the knowledge of the future outputs: $e_n(1)$ is functionally dependent on $u_n(k)$, $su_n(k)$, $k \in N$. Precisely, from (3) we get:

$$z \frac{\text{min}(z)}{\text{md}(z)} e(z) = \text{mod}(z)[u(1)(z) - su(1)(z)] \tag{43}$$

$e_n(1)$ depends on the term

$$\sum_{j=0}^{\infty} \text{mod}_j (u_n(j) - su_n(j)) \tag{44}$$

i.e., for the one step ahead prediction input, we need j-step predictions of the set point and the non-deterministic output for all positive integers j. This is a mathematical result; the physical problem is that predictions are not real values. Thus the strategy of the SISO AMAC cannot be totally ensured.

We are going to show how this mathematical solution can be used to design the control law.

To simplify the statement, we will suppose no measured disturbance and as suggested by Phillipson and MPHC applications, we take exponential smoothing for prediction:

$$F_u(z) = \frac{1 - t}{1 - tz^{-1}}$$

$$F_w(z) = \frac{1 - r}{1 - rz^{-1}}$$

(45)

with t,r called tracking or regulation coefficients. Moreover, as there is no problem on the model's denominator, we suppose an MA model (with p the number of coefficients)

$$Mc(z) = \frac{mc(z)}{z^p}$$

(46)

with mc(z) factorized in mi(z)·mo(z).

So we will work with the block representation given by Figure 9 and the AMAC law given by the relations

$$md_0 e_n(1) = - \sum_i md_i e_{n+1-i} + u_n(1) - su_n(1)$$

(47)

233

$$u_n(j) - u_n(1) = tu_{n-1}(1) + (1-r)u_n; \quad u_0(1) = u_0$$

$$su_n(j) = su_n(1) = t \; su_{n-1}(1) + (1-r)\hat{w}_n; \quad s_0(1) = \hat{w}_0 \qquad (47)$$

$$\hat{w}_n = s_n - \sum_i mc_i \cdot e_{n-i}$$
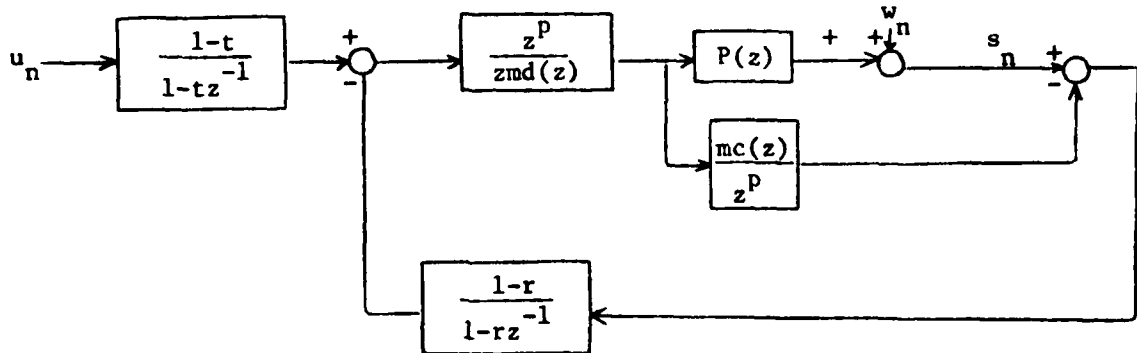
Figure 9. AMAC Representation for Study

B-3.2 Direct Application of the Mathematical Solution

From the factorization of $mc(z)$, let mi_j be the impulse response corresponding to the roots inside the circle and mod_j be the noncausal impulse response corresponding to the inverse of the factor containing the roots outside the unit circle. We write the control law as:

$$\min_0 \dot{e}_n(1) = - \sum_j \min_j \dot{e}_{n+1-j} + \sum_j mod_j (u_n(j) - su_n(j)) \qquad (48)$$

But with our constant prediction we have:

$$\min_0 \dot{e}_n(1) = - \sum_j \min_j \dot{e}_{n+1-j} + (\sum_j mod_j)(u_n(1) - su_n(1)) \qquad (49)$$

With exponential smoothing for predictions the mathematical solution gives a deconvolution model which has among the roots of the convolution model

234

only those which are strictly in the unit circle.

From (20), (21)

$$S_a(z) = \frac{1-t}{z-t} \quad \frac{mon(z)}{z^0 mon(1)}$$

$$S_{rw} = 1 - \frac{1-r}{z-r} \quad \frac{mon(z)}{z^0 mon(1)}$$

(51)

With 0 the number of roots of $mon(z)$. Thus, the prescribed behaviors can be followed only after the response time of $mon(z)$.

B-3.3   k-Step Prediction

Our mathematical presentation tells us that to compute $e_n(1)$ we need further predictions. So one idea is to rewrite the AMAC strategy as

$$s_n(k) = u_n(k)$$

(52)

with no a priori distinction between the models.

Then (47) gives

$$\sum_{i=0}^{k-1} mc_i \, e_n(k-i) = - \sum_{i=k}^{p} mc_i \, e_{n+k-i} + u_n(1) - su_n(1)$$

(53)

Thus the control $e_n(1)$ depends on the predicted inputs $e_n(j)$ and we have to solve a linear system with k unknown quantities. To get a unique solution one could introduce a criterion on the predicted inputs.

Let us look for a solution linearly dependent on the right term as:

$$\begin{pmatrix} e_n(k) \\ \vdots \\ e_n(1) \end{pmatrix} = \begin{pmatrix} f_k^{-1} \\ \vdots \\ f_1^{-1} \end{pmatrix} \left(- \sum_{i=k}^{p} mc_i \cdot e_{n+k-i} + u_n(1) - su_n(1)\right)$$

(54)

235

$e_n(1)$ is the single quantity of interest and we have the control law:

$$f_1 e_n(1) = - \sum_{i=k}^{p} mc_i \cdot e_{n+k-i} + u_n(1) - su_n(1) \tag{55}$$

Thus the implicit deconvolution model is:

$$Md(z) = f_1 + \sum_{i=0}^{p-k} mc_{i+k} \, z^{-(i+1)} \tag{56}$$

but with the static gain relation, we need:

$$f_1 + \sum_{i=0}^{p-k} mc_{i+k} = \sum_{i=0}^{p} mc_i \tag{57}$$

and necessarily,

$$f_1 = \sum_{i=0}^{k-1} mc_i \tag{58}$$

The indetermination of expression (53) is illusive. This k step prediction strategy gives the MA deconvolution model:

$$Md(z) = \sum_{i=0}^{k-1} mc_i + z^{-1} \sum_{i=k}^{p} mc_i \, z^{k-i} \tag{59}$$

Then the problem is how to choose the integer k in such a way as $Md(z)$ has all its roots strictly in the unit circle. Obviously there is at least the solution $k = p$, but then the prescribed behavior can be followed only after the time response of the process. This solution is not interesting

but the deconvolution model can be easily computed.

This k-step prediction strategy can be extended. Given a set I of positive inters, we impose:

$$s_n(k) = u_n(k), \; \forall \, k \in I \tag{60}$$

This leads as previously to a linear system whose unknown quantities are the predicted inputs. It can be solved in various ways, but the solution must give a deconvolution model satisfying the stability conditions [14].

The advantage of such an approach is in the constrained control case: let Ω be the time invariant set of admissible inputs, we write the extended k-step prediction strategy as:

$$\underset{e_n(j) \in \Omega}{\text{Min}} \; J(s_n(k) - u_n(k); \; k \in I) \tag{61}$$

where J is a criterion.

This optimization problem gives predicted inputs satisfying the constraints. Thus, one can expect to get a better behavior owing to the fact that predicted constraints are taken into account.

B-3.4   Choice from Behavior Analysis

The deconvolution model defines the tracking behavior with the following transfer obtained in a perfect modeling hypothesis.

$$S_a(z) = \frac{Fu(z)P(z)}{z \, Md(z)} \tag{62}$$

When we can take $Md(z)$ equal to $P(z)$ the set point predictor plays the

same role as the reference model in the MPHC, *so we can extend here the ideas of Rouhani* [4].

B-3.4 Pole Placement

One can impose direct pole placement. In that case, from the specified polynomial pp(z) we get the deconvolution model as:

$$Md(z) = \frac{\min(z)pp(z)}{md(z)} \tag{63}$$

because, in perfect modeling, we have

$$S_a(z) = \frac{Fu(z)}{z} \frac{mon(z)}{pp(z)} \tag{64}$$

This method is very simple when the factorization (39) is known. If not, this problem may be very difficult to solve numerically in particular when there is a great number of roots.

B-3.5 Optimization Criterion

Another natural criterion is the minimization of a quadratic distance between the expected and the actual responses to a set point sequence:

$$J(Md(z)) = \int_{-\pi}^{+\pi} \left| \frac{Fu(e^{i\theta})}{e^{i\theta}} \left(1 - \frac{M_c(e^{i\theta})}{M_d(e^{i\theta})}\right) u(e^{i\theta}) \right|^2 d\theta \tag{65}$$

where $u(e^{i\theta})$ is a specified function.

This is equivalent to a distance between deconvolution and convolution models. Thus, *if we write*

$$Md(z) = \frac{mdn(z)}{md(z)}$$

$$Mc(z) = \frac{mcn(z)}{md(z)} \tag{66}$$

The problem is to approximate the polynomial $mcn(z)$ whose roots are on both sides of the unit circle by a polynomial $mdn(z)$ whose roots are inside the unit circle and (65) can be rewritten as

$$J(mdn(z)) = \int_{-\pi}^{+\pi} \left| 1 - \frac{mcn(e^{i\theta})}{mdn(e^{i\theta})} \right|^2 dF(\theta) \tag{67}$$

with $dF(\theta)$ a positive measure and:

$$mdn(1) = mcn(1) \tag{68}$$

Such a criterion and constraints can be *computed by the Jury-Astrom* algorithm [15].

This method gives a deconvolution model which depends only on the convolution model Mc and the set point predictor (tracking coefficient in the exponential smoothing case). Those computations may be numerically easier than polynomial factorization, but have to be done again if the predictor is changed.

B-3.6   Conclusion

The deconvolution problem can be solved using a prediction strategy. This leads to a simple method but not manageable results in the k step prediction case or to more difficult computations as polynomial factorization or constrained nonlinear optimization if we want to have more manageable results.

B-4. <u>Summary</u>

We have shown that a very general implementation of a linear time invariant control law can be done by the adapted model algorithm control. This method uses five independent physical entities: two non-deterministic signal predictors which can be deduced from disturbance modelization; two mathematical representations of the process behavior which are also given by modelization; a set point predictor which can be deduced from the control law specification. The problem is complicated by the fact that one of the representations is used in a nonphysical way and thus has to be adapted by a further prediction strategy.

REFERENCES

1   E.I. JURY

"Theory and application of the z - transform method"

Wiley, 1964.

2   J.H. MAUJAN

"Advanced control of a thermal power unit"

Proceed. of the 1979 IEEE CDC.

3   W.A. WOLOVICH, P. FERREIRA

"Output regulation and tracking in linear multivariable
systems"

Proceed. of the 1978 IEEE CDC.

4   ROUHANI

"SISO constraint on input"

Progress report. S^2I. June 1979.

5   A.S. HOUSEHOLDER

"The theory of matrices in numerical analysis"

Blaisdell, 1964.

6   F.R. GANTMACHER

"Théorie des matrices"

Tome 2. Dunod 1966.

7   A.S.B. HOLLAND

"Introduction to the theory of entire functions"

Academic Press 1973.

8   J.C. DOYLE

"Robustness of multiloop linear feedback systems"

Proceed. of the IEEE CDC, Jan.1979, San Diego.

APPENDIX 1

Proof of the linear stability theorem.

Let $C_k(z), A(z), B(z)$ be three polynomials with the relation :

$$C_k(z) = (z-1)A(z) + k\, B(z) \tag{A-1.1}$$

with :_ degree of $A(z)$ greater than degree of $B(z)$

$$\_ B(1) \neq 0 \tag{A-1.2}$$

$\_ a_0$ the highest degree coefficient of $A(z)$

$\_A(z)$ has all its roots strictly in the unit circle

$\_A(z)$, $B(z)$ with real coefficients.

We will show that it exists a vicinity of zero $V(0)$ such that if k is in $V(0) - (0)$, the roots of $C_k(z)$ are strictly in the unit circle if and only if

$$k\, a_0\, B(1) > 0 \tag{A-1.3}$$

Proof : we use continuity results : the roots of a polynomial of the complex variable and the maximum of their moduli are continuous functions of its coefficients, the highest degree coefficient being taken different from 0.

So we have :

1 - For any real k, $C_k(z)$ has $(d_a + 1)$ roots if d_a is the degree of $A(z)$.

This is Appendix 1 of the report which comprises Appendix A of the whole report.

2 - For k equal to zero, the roots of $C_0(z)$ are :
the d_a roots of $A(z)$ which are strictly in the
unit circle

The simple root : equal to 1.

3 - From the preceding continuity properties, it
exists a vicinity $V(0)$ of zero such that for
any k in this vicinity, $C(z)$ has d_a roots
strictly in the unit circle.

4 - Let us study the last root.

If the modulus is greater or equal to one, the root is
real because it is alone outside the unit circle and the
coefficients of the polynomial $C(z)$ are real. So let us
consider the polynomial $C(z)$ of the real variable, its
only root greater than one exists iff :

$$C(1) \quad C(x) > 0 \tag{A-1.4}$$

for large x greater than one.

But here we have :

$$C(1) = k \ B(1) \tag{A-1.5}$$

and the sign of $C(x)$ for large x is the one of a_0 so
there is no root if :

$$kB(1) \quad a_0 > 0 \tag{A-1.6}$$

Remark : From the hypothesis on $A(z)$, the signs of
$A(1)$ and a_0 are the same.

Proof of the non linear stability theorem.

Let us take the notations :

$$E(n)^t = ((e_{n-M}-\bar{e})\ldots(e_{n-N}-\bar{e})\ldots(e_n-\bar{e}))^t$$

$$m^t = (0 \ldots 0\ m_N \ldots m_0)^t$$

$$S(n)^t = ((s_{n-N_d}-\bar{u})\ldots(s_n-\bar{u}))^t$$

$$W(n)^t = (w_{n-N_d} \ldots\ldots w_n)^t$$

$$d^t = (d_{n-N_d} \ldots\ldots d_0)^t$$

$$V(n)^t = (v_{n-L} \ldots v_{n-N_b} \ldots v_n)^t$$

$$b^t = (0 \ldots 0\ b_{N_b} \ldots b_0)^t \qquad\qquad (A-2.1)$$

$$\pi = \begin{bmatrix} \overbrace{0\ 1\ldots\ldots 0}^{M} \\ \cdot \\ \quad\cdot \\ \qquad\cdot \\ \qquad\quad\cdot \\ \qquad\qquad 1 \\ 0 \qquad\qquad 0 \end{bmatrix} M$$

$$\mathbb{P} = \begin{bmatrix} \overbrace{P_{r-N_p}\cdots P_0}^{M} \qquad 0 \\ \cdot \qquad\qquad \cdot \\ \cdot \qquad\qquad \cdot \\ \cdot \qquad\qquad \cdot \\ 0 \qquad P_{n-N_p}\cdots P_0 \end{bmatrix} N_d$$

$$Q = \begin{bmatrix} \overbrace{q_{n-N_q} \cdots q_0}^{L} & & & 0 \\ & \cdot & & \\ & & \cdot & \\ & & & \cdot \\ & & & \cdot \\ 0 & & & q_{n-N_q} \quad q_0 \end{bmatrix} N_d$$

$$t_0^t \;=\; (1 \; . \; . \; . \; . \; 1)^t$$

$$t_1^t \;=\; (0 \; . . . . . 01)^t$$

$$\bar{u} \;=\; t_1^t \; \mathbb{P} \; t_0 \quad \bar{e}$$

With : Π = identity matrix

$$M = N_p + N_d$$

$$L = N_q + N_d$$

N number of coefficients of $m(z)$

N_d number of coefficients of $d(z)$

N_r number of coefficients of $r(z)$

N_b number of coefficients of $b(z)$

with those notations, we write-the compensator relation (29) :

$$e_{n+1} = f_n \left(\frac{1}{m_0} \left[y_n + m^t (\Pi - \pi)(E(n) + \bar{e} \; t_0) \right] \right) \qquad (A-2.2)$$

-the compensator input (27) :

$$y_n = k \left[r^t (U(n) + \bar{u} \; t_0) - d^t (S(n) + \bar{u} \; t_0) \right] - b^t V(n) \qquad (A-2.3)$$

245

-and the process output (2) :

$$S(n) = \mathbb{P} \ E(n) + \mathbb{Q} \ V(n) + W(n) \qquad\qquad (A-2.4).$$

From the equality of static gains of the sensor and the reference, we have :

$$\bar{u} \ r^t t_0 = \bar{u} \ d^t t_0 \qquad\qquad (A-2.5)$$

So :

$$y_n = k \ r^t U(n) - k d^t \ \mathbb{P} \ E(n) - (k d^t \mathbb{Q} + b^t) V(n) - k d^t W(n) \qquad (A-2.6)$$

And :

$$e_{n+1} = f_n(\bar{e} + \frac{1}{m_0} \ [(m^t(\Pi - \pi) - k d^t \ \mathbb{P}) E(n) + k r^t U(n)$$
$$- (k d^t \mathbb{Q} + b^t) V(n) - k d^t W(n)]) \qquad\qquad (A-2.7)$$

Thus, we have a state representation of the control

$$E(n+1) = F_n(\mathbb{A} \ E(n) + x_n t_1 + \bar{e} \ t_0) - \bar{e} \ t_0 \qquad\qquad (A-2.8)$$

with :

$$F_n \begin{vmatrix} x_1 \\ \\ x_1 \\ \\ x_0 \end{vmatrix} = \begin{vmatrix} x_1 \\ \\ x_1 \\ \\ f_n(x_0) \end{vmatrix} \qquad\qquad (A-2.9)$$

$$\mathbb{A} = \begin{vmatrix} 0 & 1 & & 0 \\ \\ 0 & & 0 & 1 \\ \\ -\frac{m^t}{m_0}(\pi - II) & -\frac{k d^t}{m_0} \mathbb{P} \end{vmatrix} \qquad\qquad (A-2.10)$$

246

$$x_n = \frac{1}{m_0}(kr^t U(n) - (kd^t Q + b^t)V(n) - kd^t W(n)) \qquad (A-2.11)$$

Now we remark that A is a companion matrix associated to the polynomial :

$$(z-1)m(z)z^{N_p} + k\ d^t p_n(z)$$

So we have a new result of stability :

Let $\rho\ (A)$ be the spectral radius of the matrix A, if f_n satisfies the following inequality :

$$|f_n(x+\bar{e}) - \bar{e}| < \frac{k}{\rho(A)} \cdot |x| \ , \ k < 1 \qquad \forall\ n \qquad (A-2.12)$$

Then the system with the non linearity f_n is stable.

Proof :

If the linear system is asymptotically stable, the spectral radius of A is less than one, then it exists a consistent norm of A which is less than one [5] .

For that norm, we have :

$$\|A\ x\| < (A) . \|x\| \ , \ \rho(A) < 1 \qquad (A-2.13)$$

So :

$$\|A E(n) + x_n t_1\| < \rho(A)\ \|E(n)\| + |x_n| . \|t_1\| \qquad (A-2.14)$$

But from (42), we have also :

$$\|F_n(\ A\ E(n) + x_n t_1 + \bar{e}t_0) - \bar{e}t_0\| < \frac{k}{\rho(A)}\ \|A\ E(n) + x_n t_1\| \qquad (A-2.15)$$

247

So :

$$\| E(n+1) \| \leq k \left(\| E(n) \| + \frac{|x_n| \cdot \| t_1 \|}{\rho (A)} \right) \qquad (A-2.16)$$

Then if x_n is bounded i.e. set-point, measured and unmeasured disturbances are bounded, we can conclude our proof.

APPENDIX 3

Spectral analysis of the matrix $(\cos(i-j)\theta)$.

From the relation :

$$\cos(i-j)\theta = \cos i\theta . \cos j\theta + \sin i\theta . \sin j\theta \, , \qquad (A-3.1)$$

we can write :

$$\left| .\cos(i-j)\theta \, . \right| = \left| \cos i\theta \right| (.\cos j\theta \, .) + \left| \sin i\theta \right| (.\sin j\theta \, .)$$

$$(A-3.2)$$

So we see that the matrix is semi definite positive with only two positive eigen values. We are looking for the eigen vectors as a linear combination :

$$x \left| \cos j\theta \right| + \left| \sin j\theta \right|$$

We have to find x and y such that :

$$\left| \cos i\theta \right| (x. \sum_{j=0}^{M} \cos^2 j\theta + \sum_{j=0}^{M} \cos j\theta \sin j\theta) \qquad xy \left| \cos i\theta \right|$$

$$+ \qquad\qquad = \qquad + \qquad (A-3.3)$$

$$\left| \sin i\theta \right| (x. \sum_{j=0}^{M} \cos j\theta \sin j\theta + \sum_{j=0}^{M} \sin^2 j\theta) \qquad y \left| \sin i\theta \right|$$

We deduce the expressions :

$$xA+B=xy$$

$$xB+C= y \qquad\qquad (A-3.4)$$

249

with $\quad A = \displaystyle\sum_{j=0}^{M} \cos^2 j\theta$

$B = \displaystyle\sum_{j=0}^{M} \sin j\theta \; \cos j\theta$ $\qquad$ (A-3.5)

$C = \displaystyle\sum_{j=0}^{M} \sin^2 j\theta$

We get :

$$x^2 B + x(C-A) - B = 0 \qquad (A-3.6)$$

$$x = \frac{A-C \pm \left((C-A)^2 + 4B^2\right)^{\frac{1}{2}}}{2B}$$
$$\qquad (A-3.7)$$

$$y = \frac{A+C \pm \left((C-A)^2 + 4B^2\right)^{\frac{1}{2}}}{2}$$

but :

$A+C = M+1$

$$A-C = \sum_{j=0}^{M} \cos^2 j\theta = \left|\frac{\sin(M+1)\theta}{\sin\theta}\right| \; \cos\frac{M}{2}\theta$$
$$\qquad (A-3.8)$$
$$2B = \sum_{j=0}^{M} \sin^2 j\theta = \left|\frac{\sin(M+1)\theta}{\sin\theta}\right| \; \sin\frac{M}{2}\theta$$

So :

$$x = \frac{\cos\frac{M}{2}\theta \pm 1}{\sin\frac{M}{2}\theta}$$
$$\qquad (A-3.9)$$

$$y = \frac{M+1}{2} \pm \frac{1}{2}\left|\frac{\sin(M+1)\theta}{\sin\theta}\right|$$

250

APPENDIX B

APPLICATION OF ROBUSTNESS ANALYSIS TO SIMPLE MISSILE EXAMPLE

The following example applies the robustness analysis of Section III
to a single input-single output, second order model of missile longitudinal
dynamics (see Section VIII). After some preliminary analysis to obtain
the plant and model z-transform used in the robustness analysis, we
analyze the stability of the controlled plant when some parameters (plant
parameters, sample rate, reference rate, etc.) are varied.

1. Preliminary Analysis

As shown in Section III, the stability of the (sampled) controlled
plant is determined by the zeros of the expression

$$(z-1)\tilde{H}(z) + (1-\alpha)H(z) \tag{B.1}$$

where $\tilde{H}(z)$ is the z-transform of the plant model, $H(z)$ is the z-transform
of the plant, and α is the rate of the first-order output reference trajec-
tory $(0 \leqslant \alpha \leqslant 1)$. The MAC controlled system is stable if the zeros of (B.1)
lie within the unit disc. In this section we derive expressions for $H(z)$
and $\tilde{H}(z)$ in order to analyze the zeros of (B.1) in the next section.

Suppose the real plant is given by a linear, time invariant state
description

$$\dot{x} = Ax + Bu \tag{B.2}$$

$$y = Cx \tag{B.3}$$

251

If this system is relaxed at $t = 0$ (i.e., $x(0) = 0$) and if the input u is constant over the intervals $[k\Delta, (k+1)\Delta)$ of length Δ for $k = 0,1,2,\ldots$, then

$$y(k\Delta) = \sum_{\ell=0}^{\infty} Ce^{A\Delta\ell}A^{-1}(e^{A\Delta}-I)Bu_{k-\ell-1} \qquad (B.4)$$

In (B.4) it is assumed that all the eigenvalues of A have negative real parts so that the inverse A^{-1} is well defined. The z-transform corresponding to (B.4) is

$$H(z) = \sum_{\ell=0}^{\infty} Ce^{A\Delta\ell}A^{-1}(e^{A\Delta}-I)Bz^{-\ell} \qquad (B.5)$$

which is equivalent to

$$H(z) = C(I - z^{-1}e^{A\Delta})^{-1}A^{-1}(e^{A\Delta}-I)B \qquad (B.6)$$

We assume that the plant model used by MAC has a form similar to the plant (B.2), (B.3) in the following way. The model uses a discrete-time impulse response computed from a linear, time-invariant state description with matrices $\tilde{A}$, $\tilde{B}$, $\tilde{C}$ corresponding to A, B, C in (B.2), (B.3). The time interval is the same as above, namely Δ. However, the model has the following differences. The infinite sum in (B.4) is truncated after a finite number of terms. In addition, $A^{-1}(e^{A\Delta}-I)$ is approximated by $\Delta \cdot I$. With these assumptions we obtain the following expression for $H(z)$.

$$\tilde{H}(z) = \sum_{\ell=0}^{N} \tilde{C}e^{\tilde{A}\Delta\ell}\tilde{B}\Delta z^{-\ell} \qquad (B.7)$$

252

This is equivalent to

$$\tilde{H}(z) = \tilde{C}(I - z^{-1}e^{\tilde{A}\Delta})^{-1}(I - z^{-(N+1)}e^{\tilde{A}\Delta(N+1)})\tilde{B} \qquad (B.8)$$

In the examples we will assume that A, B, C have the following form as given in Section VIII.

$$A = \begin{pmatrix} Z_w & 1 \\ M_\alpha & 0 \end{pmatrix} \qquad (B.9)$$

$$B = \begin{pmatrix} 0 \\ M_{sq} \end{pmatrix} \qquad (B.10)$$

$$C = (1 \quad 0) \qquad (B.11)$$

The expressions for $\tilde{A}$, $\tilde{B}$, $\tilde{C}$ will be the same, with $\tilde{Z}_w$, $\tilde{M}_\alpha$, $\tilde{M}_{sq}$ replacing Z_w, M, M_{sq} in (B.9) - (B.11).

In order to compute H(z) and $\tilde{H}(z)$ it will be necessary to compute e^{At} for A of the form (B.1). If $\delta = \frac{1}{2}Z_w$ and $\omega = -\frac{1}{2}(-4M_\alpha - Z_w^{\ 2})^{\frac{1}{2}}$ then e^{At} is given by

$$e^{At} = \begin{pmatrix} e^{\delta t}(\cos\omega t + \frac{\delta}{\omega} \sin\omega t) & e^{\delta t} \frac{\sin\omega t}{\omega} \\ -e^{\delta t} \frac{(\delta^2 + \omega^2)}{\omega} \sin\omega t & e^{\delta t}(\cos\omega t - \frac{\delta}{\omega} \sin\omega t) \end{pmatrix} \qquad (B.12)$$

2. Examples of Robustness Analysis

2.1 Example 1

In this example, the plant parameters are $Z_w = -1.4868$, $M_\alpha = -149.93$,

$M_{sq} = -281.11$, which gives

$$\delta = -.7434, \qquad \omega = 12.2220$$

The corresponding H(z) is

$$H(z) = -(1.1807)z(z + .9490)(z^2 - (.6342)z + (.8620))^{-1}$$

which has zeros at $z = 0$ and $z = -.9490$, both stable as they should be.

The model plant used by MAC in this case will have the same parameters, i.e., $\tilde{Z}_w = Z_w$, $\tilde{M}_\alpha = M_\alpha$, $\tilde{M}_{sq} = M_{sq}$ and with $N = 49$, $\Delta = .1$. The corresponding $\tilde{H}(z)$ is

$$\tilde{H}(z) = -(2.0071)z^{-49}(z^2 - (.6342)z + (.8620))^{-1}(z^{50} + (.0280)z - (.0056))$$

Note that $z^{50} + .0280z - .0056$ can be written as

$$z^{50} + .0280(z - .2000)$$

Since $|.0280| < \tfrac{1}{2}$, Rouché's theorem (see Ahlfors, 1966) implies that $z^{50} + .0280(z - .2000)$ and z^{50} have the same number of roots in the unit circle. In other words, all zeros are inside the unit disc; hence, all zeros of H(z) are stable.

The controlled system's stability depends on the zeros of

$$(z-1)\tilde{H}(z) + (1-\alpha)H(z) = (1-\alpha)(-1.1807)z(z + .9490)(z^2 - .6342z + .8620)^{-1}$$

$$+ (z-1)z^{-49}(z^2 - .6342z + .8620)^{-1}(z^{50} + .0280z - .0056)(-2.0071)$$

The zeros of this function are the same as the zeros of the polynomial $f(z)$

$$f(z) = (1-\alpha)(z + .9490)(-1.1807) + (z-1)(z^{50} + .0280z - .0056)(-2.0071)$$

which is equivalent to

$$f(z) = z^{51}(1-\alpha+1.6999) + z^{50}((1-\alpha)(.9490) - 1.6999) + (.0476)(z-1)(z-.2000)$$

Let $f_1(z)$ and $f_2(z)$ be defined as

$$f_1(z) = (1-\alpha+1.6999)z^{51} + ((1-\alpha)(.9490) - 1.6999)z^{50}$$

$$f_2(z) = (.0476)(z-1)(z-.2000)$$

Thus, $f(z) = f_1(z) + f_2(z)$. For $|z| = 1$, it is straightforward to check that for $\alpha < 1$

$$|f_2(z)| < |f_1(z)|$$

Hence, by Rouché's theorem, $f(z)$ has the same number of roots as $f_1(z)$ within the unit disc. It is easy to see that $f_1(z)$ has all its 51 roots inside the unit disc for $0 \leqslant \alpha < 1$. Thus, the controlled system is stable for all $0 \leqslant \alpha < 1$ in this case.

2.2 Example 2

In this example the plant parameters are $Z_w = -.27877$, $M_\alpha = -64.928$, $M_{sq} = -43.285$ which gives $\delta = -.1394$, $\omega = 8.0566$. The corresponding H(z) is

$$H(z) = -(.0087)z(z - .9710)(z^2 - (1.9721)z + .9725)^{-1}$$

The model plant used by MAC in this case will have parameters the

255

same as for the previous case, $Z_w = -1.4868$, $M_\alpha = -149.93$, $M_{sq} = -281.11$ and with $N = 49$, $\Delta = .1$. As before, $\tilde{H}(z)$ is given by

$$\tilde{H}(z) = -(2.0071)z^{-49}(z^2 - .6342z + .8620)^{-1}(z^{50} + .0280z - .0056)$$

The stability of the controlled plant depends on the zeros of the polynomial $f(z)$ given by

$$
\begin{aligned}
f(z) = &(-2.0071 - k(.0087))z^{53} \\
&+ (5.9653 + k(.0140))z^{52} \\
&+ (-5.9101 - k(.0126))z^{51} \\
&+ (1.9519 - k(.0073))z^{50} \\
&+ (-.0562)z^4 \\
&+ (.1782)z^3 \\
&+ (-.1989)z^2 \\
&+ (.0877)z \\
&+ (-.0108)
\end{aligned}
$$

where $k = 1-\alpha$ and α is the parameter in the reference trajectory, $0 \leqslant \alpha \leqslant 1$. To test the stability of the zeros of $f(z)$ it is necessary to use a numerical test such as the Schur-Cohn test (see Tretter, 1976, p. 104) to test whether any zeros of $f(z)$ are outside the unit disc. For example, there are roots on or outside the unit disc if $k = 0$, $1-e^{-1}$ ($\approx .6321$), but all roots are inside the unit disc for $k = 1$.

256

APPENDIX C

DERIVATION OF THE OPTIMUM IMPULSE RESPONSE
FUNCTION FOR NONMINIMUM PHASE SYSTEMS

1. ### Derivation of the Riccati Equation for $\rho = 0$

The optimum control problem of Section IV is:

$$\xi(k+1) = U\xi(k) + \underline{b}_1 u(k) \tag{C.1}$$

$$J(n) = \tfrac{1}{2}\xi^T(n)Q\xi(n) + \tfrac{1}{2}\sum_{j=0}^{n-1} \xi^T(j)Q\xi(j) \tag{C.2}$$

Since dim $[\underline{b}_1, U\underline{b}_1, \ldots, U^N\underline{b}_1]$ is equal to N+1, the system (C.1) is controllable.

The cost function J(n) has the peculiarity of not depending explicitly on the input u'(j). But in order to be able to use the common discrete version of LQ, one must require that the cost associated with the input u'(j) be strictly positive (Sage and White, 1977; Murata, 1977). That is, the cost function must be of the type:

$$J_c(n) = \tfrac{1}{2}\xi^T(n)Q\xi(n) + \tfrac{1}{2}\sum_{j=0}^{n-1}\{\xi^T(j)Q\xi(j) + \rho u'^2(j)\}, \text{ with } \rho > 0$$

In practice it is always possible to "correct" J with an additional term $\rho\sum_{j=1}^{n-1} u'^2(j)$, where ρ is a small positive number, and obtain J_c. By minimizing J_c, one gets a suboptimal solution. Since this suboptimal solution depends on the value of ρ, one would expect that the smaller the

value of ρ, the closer the suboptimal solution should be to the optimal one.

In LQ optimal control, the intuitive significance of requiring a strictly positive input weight in the cost function J_c is to assure the boundedness of the input amplitude. But in the present problem, the input $u'(j)$ equals the first state component at time $(j+1)$: $\xi_1(j+1) = u(j)$. Therefore, the positive weight q_{11} associated to $\xi_1(j+1)$ would prevent the value of $u'(j)$ from becoming unbounded.

Motivated by the above consideration, we tackle the problem of finding the optimal input $u'(n)$, if it exists, for the case $\rho = 0$. In order to do so, we derive the Riccati matrix for $\rho > 0$ and observing its continuous dependence on ρ, we let $\rho \to 0$.

The Hamiltonian H_k has the expression:

$$H_k = \frac{1}{2}[\xi(k)Q\xi(k)] + \lambda^T(k+1)[U\xi(k) + \underline{b}_1 u'(k)] + \frac{\rho}{2}u'^2(k) \tag{C.4}$$

$$\lambda(k) = \partial H/\partial\xi(k) = Q\xi(k) + u^T\underline{\lambda}(k+1) \tag{C.5}$$

$$0 = \partial H/\partial u(k) = \rho u'(k) + \underline{b}_1^T\underline{\lambda}(k+1) \tag{C.6}$$

The conjugate equations are:

$$\rho\xi(k+1) = \rho U\xi(k) - \underline{b}_1\underline{b}_1^T\lambda(k+1) \tag{C.7}$$

$$\lambda(k) = U^T\lambda(k+1) + Q\xi(k) \tag{C.8}$$

with the boundary conditions:

$$\lambda(n) = Q\xi(n) \tag{C.9}$$

Now introduce the matrix $P(k)$ by

$$\lambda(k) = P(k)\xi(k)$$

Then equations (C.7) through (C.9) are rewritten as:

$$[\rho I + \underline{b}b_1{}^T P(k+1)]\xi(k+1) = \rho U\xi(k)$$

$$U^T P(k+1)\xi(k+1) = [P(k) - Q(k)]\xi(k)$$

$$P(n) = Q$$

Upon eliminating $\xi(k)$ and $\xi(k+1)$ from the above equations we deduce:

$$U^T P(k+1)[I + \frac{b_1 b_1{}^T}{\rho} P(k+1)]^{-1} U = P(k) - Q$$

$$P(n) = Q \tag{C.10}$$

Note that

$$[I + \frac{b_1 b_1{}^T}{\rho} P(k+1)]^{-1} = \frac{1}{\rho + P_{11}(k+1)} \begin{bmatrix} \rho & -P_{12}(k+1) & \cdots & -P_{1N}(k+1) \\ 0 & \rho + P_{11}(k+1) & & \\ & & \ddots & O \\ & O & & \ddots \\ & & & & \rho + P_{11}(k+1) \end{bmatrix}$$

Carrying out the matrix multiplication of (C.10) we deduce:

$$U^T P(k+1) \begin{bmatrix} \dfrac{-P_{12}(k+1)}{\rho+P_{11}(k+1)} & \dfrac{-P_{13}(k+1)}{\rho+P_{11}(k+1)} & \cdots & \dfrac{-P_{1N}(k+1)}{\rho+P_{11}(k+1)} & 0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} + Q = P(k)$$

(C.11)

The dependence of the above Riccati equation on ρ is throughout $1/[\rho+P_{11}(k+1)]$ which is continuous for $\rho \geqslant 0$ provided that $P_{11}(k+1) \neq 0$. That is, if $P_{11}(k) \neq 0$ for all $k \in [1,n]$ and all $\rho \geqslant 0$, we can let $\rho \to 0$ in the above equation and deduce the corresponding gain matrix $P(k)$.

Let us show that the entry $P_{11}(k)$ never vanishes, i.e., $P_{11}(k) \neq 0$ for all $k \in [1,n]$. This results from the following induction:

$$P(n) = Q \text{ implies that } P_{11}(n) = q_{11} = h_1^2 > 0 \text{ (see Section IV)}.$$

From equation (C.11) we have

$$P_{11}(k) = q_{11} + \frac{[P_{11}(k+1)P_{22}(k+1) - P_{12}^2(k+1)] + \rho P_{22}(k+1)}{\rho+P_{11}(k+1)}$$

$P(k+1)$ being a semi-definite matrix, it results that:

$$P_{11}(k+1) \geqslant 0$$

260

$$P_{22}(k+1) \geqslant 0$$

$$P_{11}(k+1) - P_{12}^{2}(k+1) \geqslant 0 \text{ (all the minors of the non-negative matrix } P(k+1) \text{ are non-negative.)}$$

Hence it follows that $P_{11}(k) \geqslant q_{11} > 0$ for all $\rho \geqslant 0$.

Now we can let $\rho = 0$ in equation (C.11) and the resulting equation (C.12) will generate the sequence $P(k+1)$ corresponding to the case $\rho = 0$.

$$P(k) = Q + U^T P(k+1) \begin{bmatrix} \dfrac{-P_{12}(k+1)}{P_{11}(k+1)} & \cdots & \dfrac{-P_{1N}(k+1)}{P_{11}(k+1)} & 0 \\ 1 & 0 \ . \ . & 0 & 0 \\ 0 & 1 \ . \ . & 0 & 0 \\ \vdots & \vdots \ \vdots \ \vdots & \vdots & \vdots \\ 0 & 1 \ . \ . & 1 & 0 \end{bmatrix} \tag{C.12}$$

2. The Optimum Input $u'(j)$

The optimum input $u'(j)$ is derived from $P(j+1)$ by:

$$u'(j) = L(j)\xi(j) = -[b_1^T P(j+1) b_1] b_1^T P(j+1) U \xi(j)$$

$$u'(j) = \frac{1}{P_{11}(j+1)} [P_{12}(j+1), \ldots, P_{1N}(j+1), 0] \xi(j)$$

Letting n go to infinity, the cost function J becomes

$$J = \frac{1}{2} \sum_{j=0}^{\infty} \xi^T(n) Q \xi(n)$$

Equation (C.12), solved backward for large n, converges to a "steady state" Riccati solution and the gain $\hat{L}(j)$ becomes time invariant:

$$\hat{L} = \frac{1}{P_{11}} [P_{12}, P_{13}, \ldots, P_{1N}, 0]$$

Since only the first column of the matrix P is of importance for the determination of $\hat{L}$, we can write the Riccati recursive equation in terms of column vectors of P.

$$P = [p_1 \vdots \quad \vdots p_N] \text{ and } Q = [q_1 \vdots \quad \vdots q_N]$$

Then:

$$p_1(k) = U^T p_2(k+1) - \frac{p_{12}(k+1)}{p_{11}(k+1)} U^T p_1(k+1) + q_1$$

$$\vdots$$

$$p_j(k) = U^T p_{j+1}(k+1) - \frac{p_{1(j+1)}(k+1)}{p_{11}(k+1)} U^T p_1(k+1) + q_j$$

$$\vdots$$

$$p_{N-1}(k) = U^T p_N(k+1) - \frac{p_{1N}(k+1)}{p_{11}(k+1)} U^T p_1(k+1) + q_{N-1}$$

$$p_N(k) = q_N$$